Lee *et al. EURASIP Journal on Advances in Signal Processing* 2012, **2012**:99
http://asp.eurasipjournals.com/content/2012/1/99

EURASIP Journal on Advances in Signal Processing
a SpringerOpen Journal

RESEARCH Open Access

Human tracking with an infrared camera using curve matching framework

Suk Jin Lee[1], Gaurav Shah[1], Arka Aloke Bhattacharya[2] and Yuichi Motai[1*]

Abstract

The Kalman filter (KF) has been improved for a mobile robot to human tracking. The proposed algorithm combines a curve matching framework and KF to enhance prediction accuracy of target tracking. Compared to other methods using normal KF, the Curve Matched Kalman Filter (CMKF) method predicts the next movement of the human by taking into account not only his present motion characteristics, but also the previous history of target behavior patterns-the CMKF provides an algorithm that acquires the motion characteristics of a particular human and provides a computationally inexpensive framework of human-tracking system. The proposed method demonstrates an improved target tracking using a heuristic weighted mean of two methods, i.e., the curve matching framework and KF prediction. We have conducted the experimental test in an indoor environment using an infrared camera mounted on a mobile robot. Experimental results validate that the proposed CMKF increases prediction accuracy by more than 30% compared to normal KF when the characteristic patterns of target motion are repeated in the target trajectory.

Keywords: object tracking, far-infrared imaging, human target, curve matching, Kalman filter

1. Introduction

Kalman filter (KF) is a set of mathematical equations that update the underlying system state with the repetition process of prediction and correction to minimize the estimated error covariance based on the last state for its predictions and not the history of the target motion in itself [1]. Each target motion may have a prominent moving pattern, or a cyclic motion pattern, considered as basis in [2-4]-which may not be detected accurately by KF alone, even if these characteristic patterns are repeated many times in the target trajectory. Such a repetition may provide computational benefits when it is integrated with more advanced and complex motion analysis systems.

We assume in this article that such a repetition of certain characteristic patterns exists in the target trajectory. The more the time elapsed, the greater the characteristic motion pattern which has been recorded in the past trajectory repeated. Noting and acquiring these distinctive marks can yield a more accurate prediction performance, compared to any standalone implementations of KF. We set up an experimental environment indoor to demonstrate the prediction accuracy of the developed algorithm. The experiment setup includes a human-tracking mobile robot used in an indoor environment, fitted with an infrared camera and a laser range finder.

The main contribution of this article is to implement the Curve Matched Kalman Filter (CMKF) by using curve matching framework for checking the repetition of characteristic motion patterns. This article takes KF as a basic implementation model, and shows through experimental results how curve matching can improve its prediction accuracy when it is used for the target tracking of human subjects. By combining KF with a curve matching framework, we are essentially increasing the number of states upon which the prediction process depends. Therefore, this article may provide a state-of-the-art target tracking method with the development of a filter design which improves the prediction accuracy of any filter aimed at tracking human subjects.

The rest of this article is organized as follows. In Section 2, we present related article pertaining to this study. In Section 3, we present the proposed human tracking using CMKF. Section 4 describes the experimental setup

* Correspondence: ymotai@vcu.edu
[1]Department of Electrical and Computer Engineering, Virginia Commonwealth University, Richmond, VA, USA
Full list of author information is available at the end of the article

prepared to test the new concept and presents experimental results of target tracking with different target conditions of multiple trials, compared to other methods including KF, arbitrate OFKF, particle filter, and optical flow. In Section 5, we present the prediction analysis of improvement after using CMKF and show a further application where the CMKF far outperforms a normal KF. Finally, Section 6 concludes this study and describes future work.

2. Related studies

There are two categories of articles related to the study reported in this article. The first group of articles is concerned with tracking targets with the help of mobile robots with or without the use of KF. The second group of articles deals with curve matching. They describe how to match curves efficiently and show how these concepts have been applied to target tracking in the past.

2.1. Human tracking by mobile robot

Human tracking by mobile robots has been an area of intense research for many years. The human body can be tracked by several methods including processing 2D data or 3D positional data by using normal KF [5,6], by segmentation of the main target from the background [7], by multi-sensor fusion data and laser-based leg-detection techniques using on-board laser range finder [8], by techniques such as the hierarchical KF [9], or by the use of quaternions [10]. Humans have also been tracked by tracking multiple features [11], or by localizing the position of the robot itself [12], with respect to its environment. There has been work which deals with detecting and classifying abnormal gaits [13], and human recognition at a distance through gait energy image [14], gait detection using feature extraction from spatial and temporal templates [15], or identification of humans through spatial temporal silhouette analysis [16]. None of these algorithms implement any technique where the target's trajectory is marked and learnt.

In works that have implemented the target trajectory, robots have been trained to learn from human motions as in [17,18], and to follow the way a human walks [19]. In other cases, robots have been made to model unknown environments as in [20]. However, these behavior analyses for human motion have not been specifically implemented in any prediction algorithms of target's trajectory.

2.2. KF-based tracking

Extended KF has widely been used for tracking moving people [21,22]. We have further developed the extended KF into an interactive multiple model [23,24] successfully. The KF provides a general solution to the recursive minimized mean square estimation problem within the class of linear estimators [1,25-27]. Use of the KF will minimize the mean squared error as long as the target dynamics and the measurement noise are accurately modeled. Consider a discrete-time linear dynamic system with additive white Gaussian noise that models unpredictable disturbances. The problem of this KF framework is to assume the target behavior follows a random variable distribution [1], which may degenerate the prediction accuracy of human localization. For example, acquiring the characteristics of human motion by curve matching has not been implemented in any of the aforementioned articles.

2.3. Curve matching

Curve matching has often been implemented in robotics but to a different end. Most of the works on curve matching deal with identifying some known curve from an image, and then initiating various algorithms to follow it [28-34], while [35] deals only with a parameter-free approach to register images using elasticity theory. Further work deals with how best to find out contours from an image to identify and classify a target [36-41].

Another extensive area of work has been the field of curve matching itself. There have been many articles which deal with curve matching using splines [42-46]. Other algorithms for curve matching use polygonal arc methods [47], matching the curve structure on the most significant scales [48], unary and binary measurements invariant for curves [49], fuzzy logic [50] or Sethian's Fast Marching method [51].

CMKF concentrates on using curve matching for checking motion repetition characteristics and partially bases its curve matching algorithm from [52]. All of the mentioned articles allow curve matching at a high computational cost. A new and straightforward model has thus been developed in this article, which not only improves the performance of the KF, but also keeps the time required for the extra computation as low as possible.

3. Proposed human tracking using a CMKF

3.1. Concept

This section proposes a new model called the CMKF. The model intends to track the movements of a human by essentially increasing the number of states on which the prediction depends. This approach uses mathematical curve matching to relate the motion of the human at the present instant to some similar motion in the past, and thus make a better prediction of the man's state.

We consider two curves-P denoting the man's trajectory in the past or the past curve, and C denoting the man's current trajectory or the current curve. We search in P for stretches where the present curve C matches. When a strong match is found we try to predict the next movement of the man based on how it had behaved in the past. At each instant of time, a weight is assigned to the curve matching algorithm (CMA) and the KF depending on whether the present motion of the man strongly resembles his past motion. The weighted mean of the CMA and the KF is then calculated, according to which the mobile robot is moved. To facilitate finding matches of curve *C* in curve *P*, we encode them as strings *C*' and *P*', respectively, and apply computationally light string matching algorithms to calculate the extent and degree of matches.

3.2. Justification for using curve matching

In the experiment setup, we consider a mobile robot always centered in on the human target. Whenever the human makes a move, the robot head moves accordingly to bring the human back to the center of its sensor image. Under such situations, if the human target suddenly changes direction, the KF cannot catch up immediately because it cannot predict the target using the last few states that the body was going to make such a move. Under more general circumstances, it is impossible to predict if a human will make a sudden move from his present position. However, an alternate solution is to utilize past records of motion to help predict complicated movements of human subjects. That is why we extend KF to the area of mathematical curve matching in order to increase the state-space which may be used for prediction making.

It is also noted that current trajectory may not always significantly match any motion in the past. Hence, the CMA prediction should only be used when there is ample evidence that a repetition in trajectory is taking place. The algorithm will supply the robot with a weighted average of the KF prediction as stated above and the value predicted by CMA. The better the present motion curve matches one in the past, the higher the weight of the prediction returned by the CMA. Also converting the curves into strings ensures lightweight computation, and hence viable for implementation in real-time systems. In the following sections, we describe in more detail each of the aspects of the CMA.

3.3. Converting the curve into a string

Two curves are initially calculated. The first curve *C* denotes the current motion of the human, and the second curve *P* denotes his past motion. Each curve is taken and converted into a corresponding string denoted as *C*' and *P*', respectively. These two strings are then matched. The length of *C*' represents the length of the time through which the present motion has been following a past motion. The string matching algorithm also calculates the extent to which the strings have matched. Initially, both *P*' and *C*' are null. As the robot starts tracking, the motion of the human gradually gets started to store in *P*'. Thus, *P*' gives the entire history of the motion which the robot has tracked till the present moment. *C*' stores only that much amount of the ending portion of *P*' that can be matched to an earlier instant in *P*'. Thus, *C*' only contains the last few frames of data which can be matched to a previous motion of the man. If the current motion does not match any previous motion already stored in *P*', we re-initialize *C*' to null.

We choose to encode the point-wise variation of the curve. Let us consider a motion Γ(*s*). The curve is sampled at specified intervals. The relative difference of the neighboring sample points in the graph is calculated (as shown in Figure 1). The value obtained at each sample point is then converted into an integer. A sequence of integers denoting every sampled portion of the curve is thus obtained.

The advantage of encoding the difference between successive sampled points is that similar curvatures with different range of values will be encoded in a similar manner. The curves will match not only in those portions where both the position and movement characteristics of the body is exactly the same, but also those portions where only the curvature of motion is the same. We thus have

$$F_i = \Delta\Gamma(s_i) = \Gamma(s_i + \Delta s) - \Gamma(s_i) \tag{1}$$

where F is the string of integers to which the curve is being converted, and *i* denotes a particular sampled point corresponding to a particular character in the string.

To make the method more robust, we may compute an averaged difference

$$F_i = \mathrm{Av}\Delta\Gamma(s_i) = \frac{1}{k}\sum_{j=0}^{k-1}\Delta\Gamma(s_i + j\delta) \tag{2}$$

over some fixed *j* and *k*. δ is some fixed non-negative real number.

The result is then multiplied by some experimentally decided integer *M* to space out the values obtained. We then convert each F_i to the nearest integer.

$$F'_i = \mathrm{int}(MF_i) \tag{3}$$

int is the integer function, which finds the greatest integer lesser than or equal to the number.

The sequence $F'_1F'_2,\ldots,F'_k$ thus becomes the desired string encoding of the motion curve.

Note that many special features of the curve, such as rotationally invariant points, etc., need not be taken for our purpose of curve matching. This procedure helps reduce time taken for the algorithm to run.

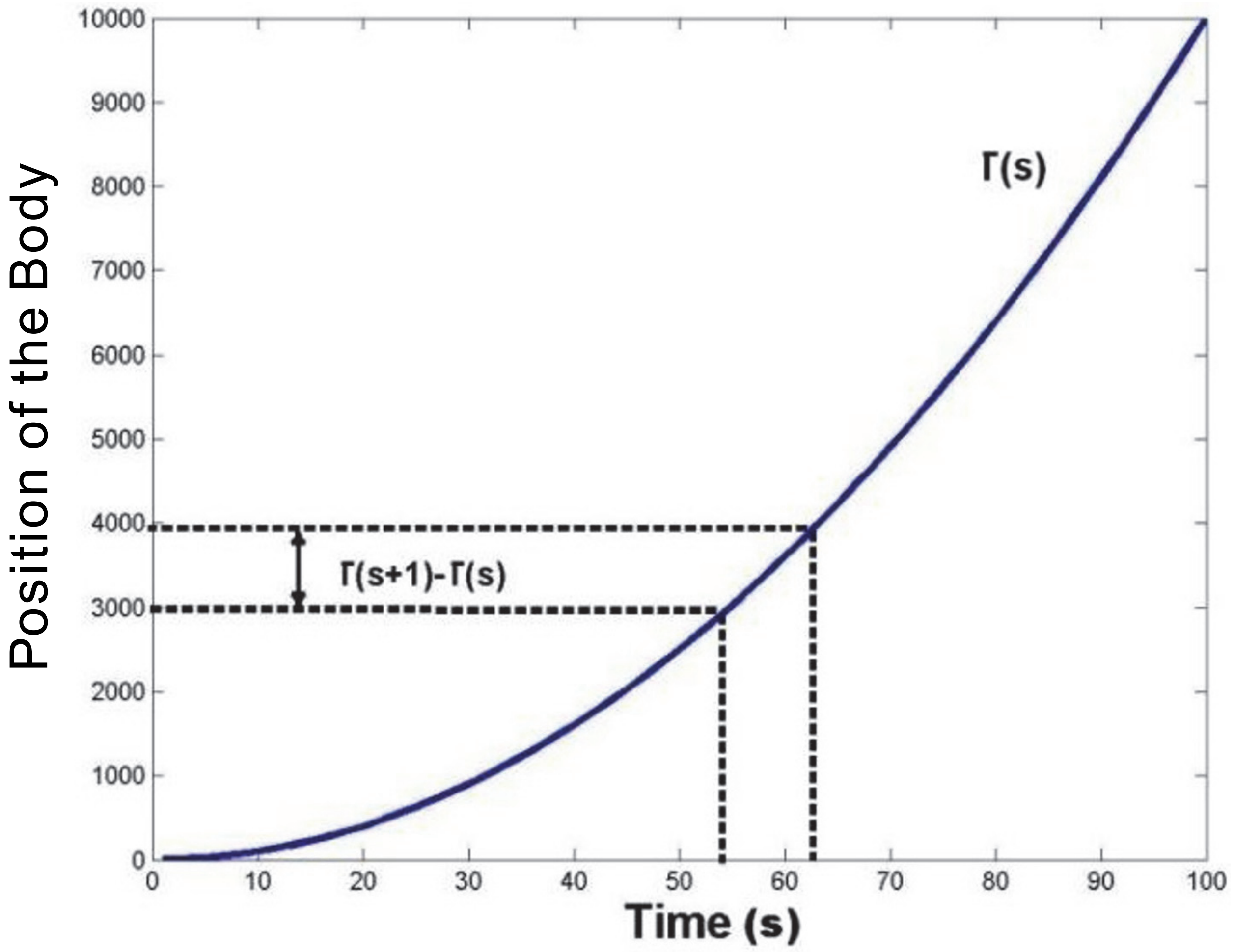


Figure 1 Method showing how to sample various points from a sample curve to convert into a string.

3.4. Matching the two strings

During the string matching, the possibility of the data being noisy should be kept in mind. Hence, we keep an accuracy factor ε. There is a match between corresponding characters if

$$P_k - C_k \leq \varepsilon \qquad (4)$$

where *P*' and *C*' are the two strings denoting the previous history and the current motion, respectively, and *k* and *k*' are the indices where they match.

3.5. Update of the strings

Initially, the string *C*' includes the latest move only. *P*' and *C*' are then matched and the points where they have matched are stored. The next move is checked at exactly these points in the previous memory string *P*'. If there is a match with the next move as well, *C′* (which denotes how much the present move has matched some action in its previous history) is augmented to include the present move, otherwise *C*' is re-initialized to contain the latest character only.

Whenever a portion of the string is matched, the curve matching prediction returned is the immediate next move taken by the human at the point where the curve has been matched. If there is more than one place where the present motion curve has matched, then we take the average of the readings. Also, as will be seen in part E, the weight of the CMA is automatically reduced if the body has been shown to act differently after each such matching motion.

Note that if the augmented *C*' curve has to match, it will match only in those indices where the unaugmented *C*' had previously matched. This way, the entirety of *P*' need not be scanned for a match every time the current string is updated. Figure 2 demonstrates how much of the present move will be matched to the previous motion. The present move will constitute *C* and the remaining curve will constitute *P*.

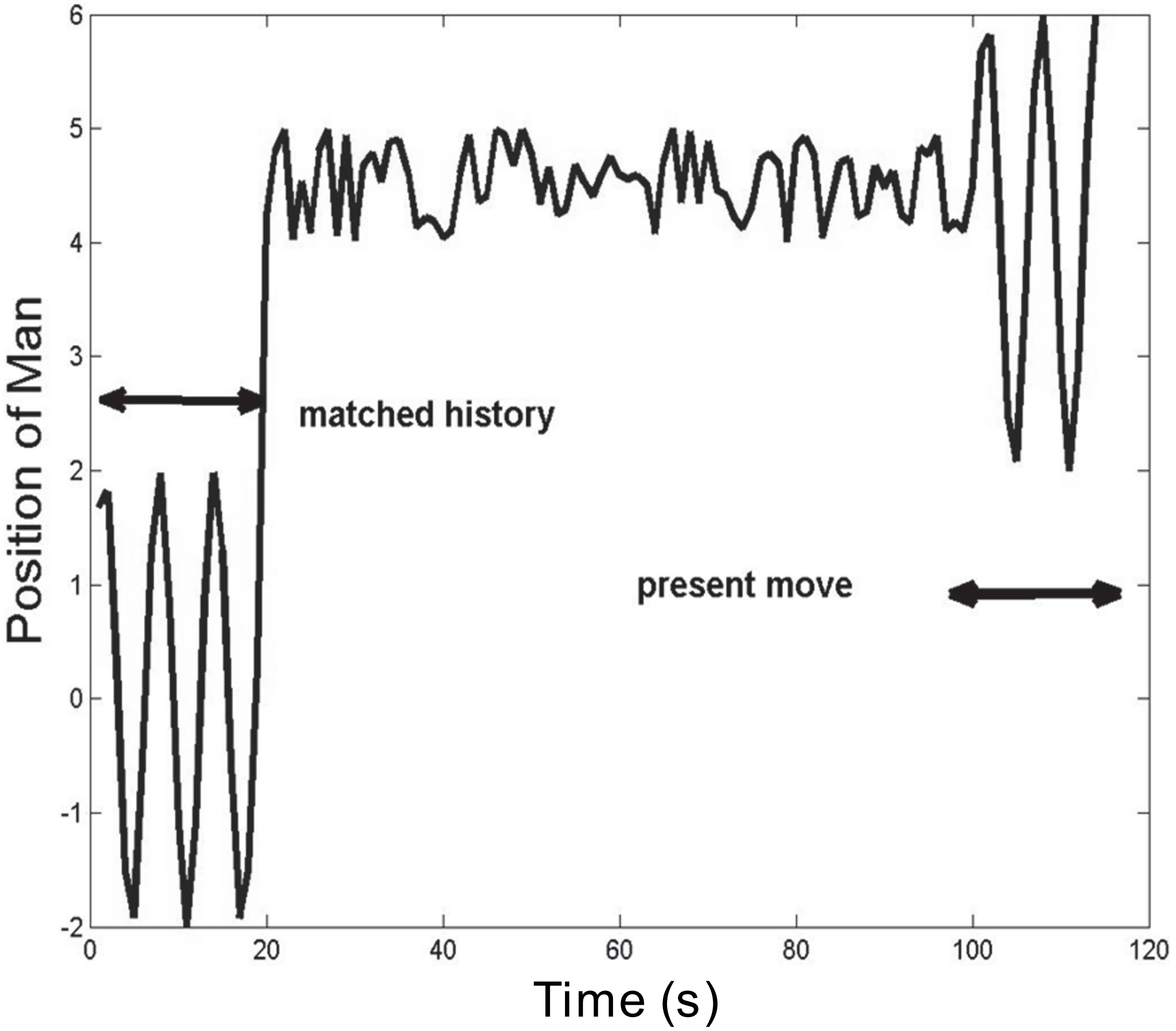


Figure 2 Demonstration of how past memory helps to predict the future motion. The body shows a tendency of a sudden jerk when it shows the present kind of behavior. With the CMA, we can predict the jerk, which we would not have been able to do with a KF.

3.6. Complexity of the CMA

Curve matching is computationally expensive. There have to be a number of algorithmic procedures which need to be adopted in order to lessen the computation required for curve matching.

Let us assume the length of *P*' to be *n*. We assume that character comparisons take place in a constant time c_1. The first value of *C*' is matched in *O*(*n*) time, by a naïve string matching algorithm. Thus, the first step has a complexity of

$$T(n) = nc_1 = O(n) \tag{5}$$

where *T*(*n*) is the time taken to run. Let the first character of *C*' match in *p* separate locations in the recorded database. The next time a character comes in, we only need to check it against the *p* positions, and not all of the *n* positions in the database. $p \le n$. The time required per iteration (as only one new data is obtained in one iteration) now is

$$T(n) = pc_1 = O(p) \tag{6}$$

The value of *p* is non-increasing for a particular match. As more of *C*' is matched, the number of positions in *P*', where it has matched will get reduced. Generally, *C*' will initially match in only some select indices in *P*', hence, it can be assumed that the entire process will require less time per cycle, even in the worst case. The time taken is an important metric as this algorithm has to be online.

For improvements in running time when *P*' becomes large, *P*' is made to contain only a limited amount of history (such as only those portions of the motion which have been matched repeatedly). The amount of history which is to be stored can be optimized based on the frame rate of the sensor used, i.e., the time required to make the CMKF prediction should not exceed the frame rate of the sensor. This will help in restricting the continuous increase in the length of the stored string *P*', and also help reduce the running time.

3.7. Assigning a weight to the CMA

Three cases need to be taken into account when deciding the weight of the prediction from the CMA.

Case (i) : Length of the match

The first aspect is the extent to which the curve matches exactly. This is very important, as this will govern whether exactly such a motion has indeed taken place in the past. If the curve only matches to a very small extent, then it will be evident that the present motion is not a strong repetition of its past motion.

This can be determined by the number of characters the two strings have matched exactly, and number of characters that were matched with some error.

$$w_{CMA} \propto \frac{l_{C',exact}}{l_{C',total}} \times f(l_{C',total}) \tag{7}$$

where w_{CMA} is the weight of the CMA, $l_{C',exact}$ is the number of characters in C' that has matched exactly to the corresponding characters in *P*', and $l_{C',\ total}$ the total number of characters in *C*'. If $l_{C',\ exact}$ equals $l_{C',\ total}$ then the curves have matched exactly, and the weight given to the CMA is high. The function *f* is a function which has a value close to 1 when $l_{C',\ total}$ is high. This is done to give more importance to longer matched curves. The function *f* can be chosen to be any suitable function, such as the sigmoid function.

Case (ii): Frequency of the match in history

The second aspect to be taken into consideration is the number of times *C*' has been matched in *P*'. There may be a case where the present motion has been repeated more than once in the past.

The more times the curve has matched, the more predictable and characteristic the behavior of the human is, and lesser is the error which the prediction is liable to have. Also, it may happen that the curve has matched just a few times but for long periods. In all such cases, the weight of the CMA is assigned to a high value.

$$w_{CMA} \propto 1 - \frac{1}{l_{C',total} \times (m+1)} \tag{8}$$

where *m* is the number of matches. As the values of $l_{C',\ exact}$ and *m* increase, the weight of the CMA also increases.

Case (iii): Motion maintenance after similar trajectory

The final aspect that needs to be investigated is motion maintenance after similar trajectory, i.e., whether or not, after similar curvature of motion in the past, the immediate next action of the human was the same. If there is more than one type of action after a matched curvature, then we know that even after a similar trajectory in the past, the body does not behave predictably. So, the weight attached to this reading has to be lessened. The weight will be attached based on the number of instances the body has taken one particular motion or the other after a certain motion. In our experiment, we always try to center the robot on the human. Hence, the human is always located at the center in the robot point of view. Any movement to any direction is either along the positive direction with respect to the origin of the robot, or to the negative direction. Hence, we can express the weight as

$$w_{CMA} \propto \delta\left(\frac{a_+ - a_-}{a_+ + a_-}\right) \tag{9}$$

where a_+ is the number of actions in the positive direction (according to the chosen reference), and a_- is the number of actions in the negative direction, following matched motions. Here, delta function δ(·) has the value as follows: 1 for δ(0); otherwise |*a*| for δ(|*a*|). Note that the weight is the highest if there is a motion maintenance, and is low in cases of inconsistent behavior.

3.8. Flowchart of entire process

CMKF consists of taking the KF prediction, the CMA prediction and assigning a weight to each of them, depending on their importance to the context. Then, the weighted average of their predictions is taken to move the robot. The entire process is described by the following six steps:

Step 1 : The human's position is calculated using sensors.
Step 2 : The KF prediction is obtained.
Step 3 : The CMA prediction is obtained.
Step 4 : The weight of the CMA is calculated as the product of the three terms shown in cases (i), (ii), and (iii).

$$w_{CMA} = k \times \frac{l_{C',exact}}{l_{C',total}} \times f(l_{C',total}) \times \left(1 - \frac{1}{l_{C',total} \times (m+1)}\right) \times \delta\left(\frac{a_+ - a_-}{a_+ + a_-}\right) \tag{10}$$

w_{CMA} can have a maximum value of 1 (the value of the constant of proportionality *k* is adjusted in the appropriate manner), and that too in the most ideal case. The weight attached to the KF is

$$w_{KF} = 1 - w_{CMA} \tag{11}$$

where w_{KF} is the weight of the prediction of the KF. It is evident that in a majority of the cases the weight of the KF far outweighs the weight of the CMA. It is only in those cases that the motion is highly repetitive that w_{CMA} will be very high.

Step 5 : Make the robot move according to the weighted average of the KF prediction and the prediction based on CMA.
Step 6 : The position of the human is obtained, the Kalman matrices are updated and the entire process beginning from Step 1 is repeated.

4. Experimental results

Section 4.1 shows the experimental environment setting. Sections 4.2, 4.3, and 4.4 show the advantages of CMKF over the KF by analyzing in detail an instance where the mobile robot was made to track a human in an indoor environment for about 400 s. Other runs, with various degrees of motion repetition, are analyzed in Section 4.5. Finally, computational time is evaluated for the complexity in Section 4.6.

4.1. Experimental setting for target tracking

We implemented a human-tracking mobile robot in an indoor environment, with an infrared laser and a laser range finder mounted on the Active Media robot. Online video was collected using Raytheon Infrared Camera operated at 15-Hz frame rate and image size of 640 × 480 pixels. The video stream was processed at 15 Hz using a PC for a real-time analysis. We claim that since the CMKF gives an improved performance on this version of the KF, it will similarly improve the performance when implemented in conjunction with other varieties of KFs, such as the use of different dynamical models. This is because the concept developed is independent of the efficiency and type of the KF implemented. We try to predict the angle by which the robot should be turned so as to bring the man back to the center if and when the man makes some motion. The range finder is used to find the distance of the man from the robot so as to estimate the relative angular position of the human accurately and also to ensure that the robot remains within a certain specified distance of the man.

Figure 3a shows a still image captured by an infrared camera, which is then thresholded to obtain only the salient part of the image. The brightest column is taken to be an approximation of the center of the human being. Figure 3b shows the relative angle between the human and the robot by making use of the range finder sensor which is mounted on the robot. Figure 3c shows how the robot is rotated about its vertical axis to keep the human at the center of the image. The piecewise constant acceleration model, or discrete-time double integrator, has been implemented for the KF. It is assumed that within two consecutive frames received by the camera, the acceleration of the human remains constant. The state vector contains information about the angular position and the velocity of the man. The state vector is updated by the piecewise constant acceleration model, the value of acceleration being calculated from the last few states, such as two consecutive frames based on the measurement data.

4.2. Target tracking by CMKF

Figure 4 shows the performance of the filter when it has to predict the current position of the man. The ground truth taken is the actual angular position of the human at the $(t + 1)$th second, and it is compared to the prediction of the CMKF made from data till the tth second. In Figure 4, the ground truth is labeled as the original curve. Once the prediction is done, we update the CMKF equations by the true angle subtended and then rotate the robot by the CMKF prediction to center on the human. We also store this value in the motion history of the target, i.e., in P'.

It may be argued that even if the human repeats his motion, his angular positions may not be exactly the same as they were previously. Moreover, the camera may have been rotated through slightly different angles now. Hence, the ground truth that we have taken to update our equations and augment P' may also change slightly. However, the variation in these values will never exceed a certain threshold which is bounded by the average error which the implemented KF has. Thus, choosing the ε in Equation (4) based on the average error of the version of the KF implemented helps us circumvent the problem.

As we can see in Figure 4, the filter worked throughout the period of study. For a certain period of time, the motion curve of the human matches to a large extent a previous motion curve. The CMKF gave much more importance to the curve matched prediction than the KF prediction during that point of time (as evident from Figure 5). The result is that the motion of the human including the jerks in this portion of the curve is predicted extremely accurately.

Figure 5a shows an example of the target behavior with the length of the current motion string, C', versus time. One of the most important criteria on which the weight of the CMA depends is the length of C'. If the present motion matches for a long period of time, the weight of the curve matching becomes high. Figure 5a shows the portion of curve matched at any instant of

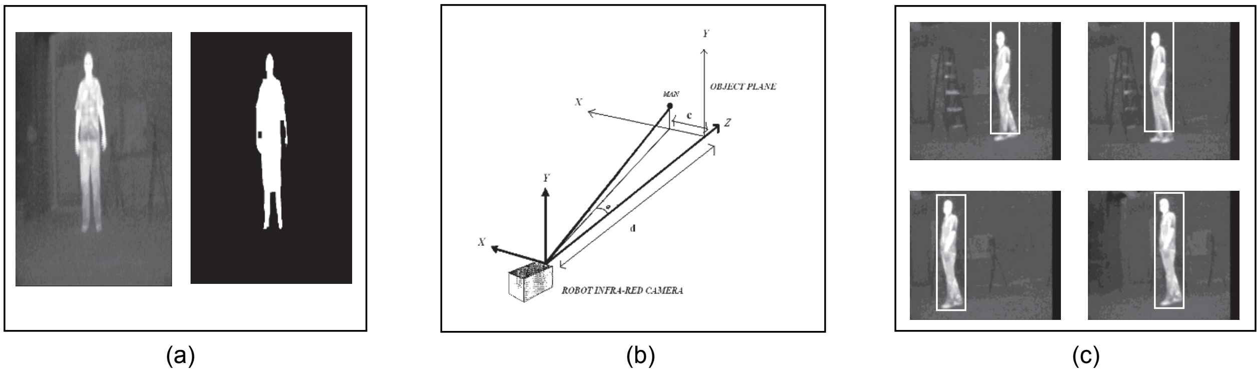


Figure 3 Experimental setup. **(a)** The infrared image obtained by the robot is thresholded to include only the salient parts of the image. **(b)** The relative angle with respect to the camera is measured. **(c)** The rotating action of the robot in the experiment. A sequence of frames where the human moves from right to left, and the robot is adjusted by the corresponding angle to bring the human back into the center of the frame.

time. It may be noted that around the time period of 16 s, the present motion matches some past motion for a long period of time. During this period of the corresponding time the weight of the CMA dominates over that of the KF, as is seen in Figure 5b.

4.3. Comparison of error between CMKF and KF

The CMKF combines the best of the KF and the CMA. As seen from Table 1 it provides almost 12% improvements in the mean error. This effect is most pronounced when the error is studied at points where the motion matches to a great extent. As shown in Figure 6, in the instances where the human repeated his motion for a long period of time there was no improvement in the prediction error due to the KF. The CMKF, on the other hand, assigns a high weight to the CMA, and predicts in accordance to how the human had behaved earlier while undertaking a similar motion.

In Figure 6, we find that the error in the portion in which the curves match is reduced by close to 95%. This results in extremely accurate tracking, and also brings down the entire mean error of the process.

4.4. Comparison of error among other methods using different target conditions

We have compared the tracking error of the proposed CMKF with other methods including optical flow, arbitrate OFKF, and particle filter, as shown in Figure 7.

In Figure 7, we find that optical flow has a significant angle error at the beginning of the prediction and that the particle filter also has a significant angle error at transient states. The angle errors of arbitrate OFKF are fluctuating over all the observation periods. The standard deviations of angle error for optical flow, arbitrate OFKF, particle filter, and CMKF are 2.25, 1.93, 3.01, and 0.25, respectively, as shown in Figure 7.

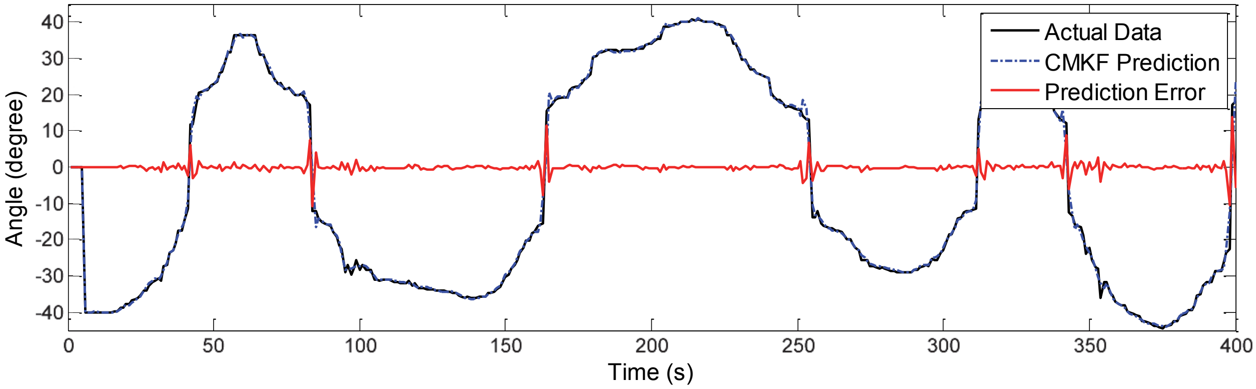


Figure 4 Plot of estimated angle by the proposed CMKF and actual angle versus time. We can see that the two curves are almost indistinguishable. Though there is some slight error at certain junctures, as soon as the motion of the man is repeated to a sufficient extent, the proposed filter gives almost an exact prediction.

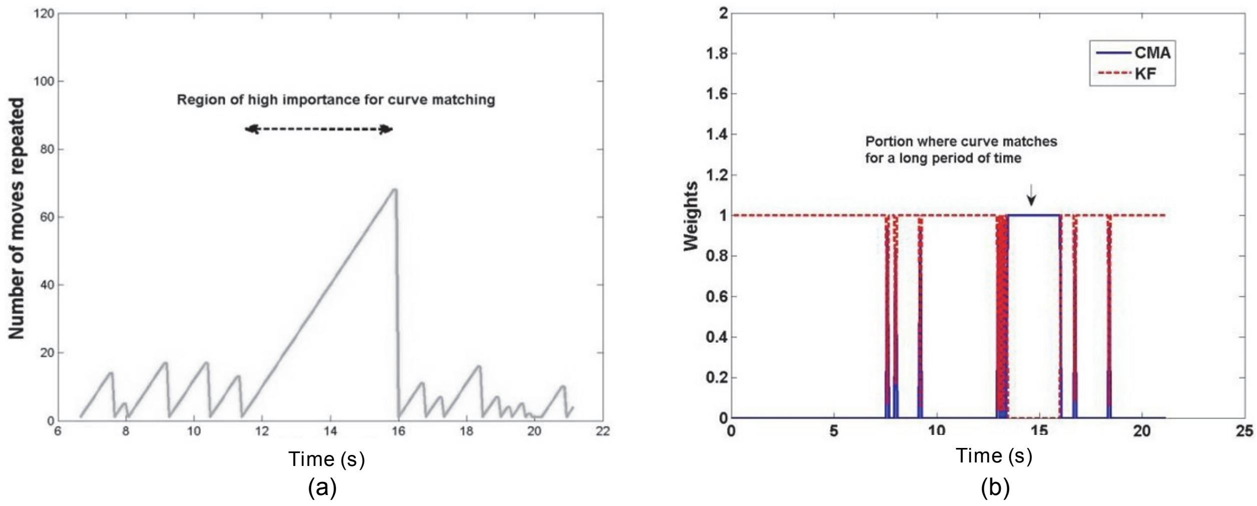


Figure 5 Curve matching algorithm analysis. **(a)** The number of latest moves which has matched against the past motion at that instant of time. The middle portion matches to a large extent and hence the CMA prediction was given importance over the KF prediction, resulting in the exact track. **(b)** The corresponding weights of the two filters-the KF and the CMKF at any instance of time during the tracking. It may be noted that the weight of the KF remains high almost throughout, except in the portions where the CMA finds a repeated trajectory in the motion of the human.

We have evaluated consistency of target prediction in comparison to the duration of the tracking correctness with different tracking condition. As shown in Table 2 the proposed CMKF shows the best performance for the duration of the tracking correctness within 3-sigma bound. We have also checked the convergence using standard deviation of the tracking error under different conditions (ground proof and noisy) with variable motion types (fast, medium, and slow). As the value of the standard deviation is small, it is expected to reach the small convergence. As shown in Table 2 the proposed CMKF shows the smallest value among other tracking filters. We finally evaluated the degree of bias for residual error during the stable periods. As shown in Table 2 the proposed CMKF presents very small bias for residual error during stable periods, where we set up the stable periods from 300 to 400 sample states.

4.5. Multiple trials with Monte Carlo simulation

We have evaluated the performance of the proposed CMKF with multiple trials with Monte Carlo simulation, where we used different speed target movement, such as slow, medium, and fast, as shown in Figure 8. In the fast speed movement, we notice outstanding overshoots around the observation period, whereas both slow and medium speed movements show small angle error within 1.5° overall the observation period. We summarized the results including mean error, standard deviation, and mean time required per cycle in Table 3. As shown in Table 3 the proposed CMKF works better in the fast speed target movement than in slow and medium speed target movements.

Figure 8 shows the comparison of tracking error among the multiple trials of different speed target movements. In the fast speed target movement, we find that some significant tracking errors at the change of degree acceleration, but the overall angle errors of multiple trials of tracking are within 1.5° with different speed using CMKF. We summarize the mean error, the standard deviation of error, and mean time required per cycle in Table 3.

Figure 9 shows the comparison of percentage duration of tracking correctness including ground-proof, human target-only, and noisy-environment with multiple speed trials of target movement. The average percentage durations of tracking correctness for CMKF are 45.50 for ground-proof, 34.61 for human target-only, and 40.75 for noisy-environment. The average percentage durations of tracking correctness for particle filter are 34.50 for ground-proof, 33.35 for human target-only, and

Table 1 Details of tracking

	Mean error	Standard deviation of error	Mean time required per cycle	Duration of the tracking correctness within 3-sigma bound (%)
KF	0.4868	5.2748	1.5992e-4	15
CMKF	0.0697	0.0662	7.2918e-5	32

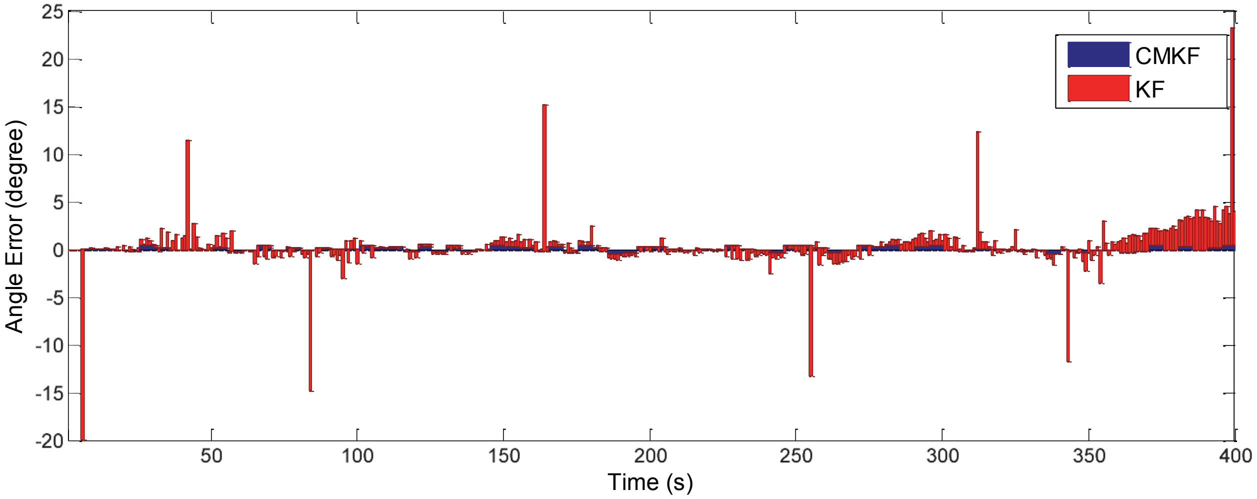


Figure 6 The two bar graphs shown are the magnified graphs for the region when the action is repeated to a large extent. The first bar graph shows the error in prediction of the angular position of the man when he is tracked by the CMKF. The bar graph on the right shows the error predicted when the man is tracked by only the KF. We can see that the error is much lower in the CMKF during the time when the man tends to be repetitive in its motion.

35.83 for noisy-environment. The average percentage durations of tracking correctness for arbitrate OFKF is less than 10% over all the conditions, whereas the optical flow has very few tracking correctness over all the conditions.

4.6. Comparison of time complexity

It has also been determined that the time required by the proposed algorithm to make its prediction from a particular frame does not differ significantly from the time taken by the normal Kalman prediction algorithm. Due to the efficiency of the algorithm proposed, the extra computations are required for the CMAs.

As shown in Figure 10, the consecutive peaks at intermediate points are due to the start of the curve matching section of the CMKF algorithm. Many initializations and preliminary calculations take place, and so the time required for this particular step is higher than the rest. It is still within the order of 10e-4 s, so there is no observable delay.

We summarized the average mean time with other methods in Table 4. The proposed CMKF requires more time complexity in comparison with KF, arbitrate OFKF, and optical flow, but it showed much better prediction accuracy than the others. We also find that CMKF needs less time complexity than particle filter even though CMKF shows better prediction accuracy than particle filter.

5. Prediction analysis via evaluation

5.1. Analysis of improvement after using CMKF

The algorithm works when tracking humans with a tendency of repetitive motion. Since each human has his own characteristic motion, given a long enough track

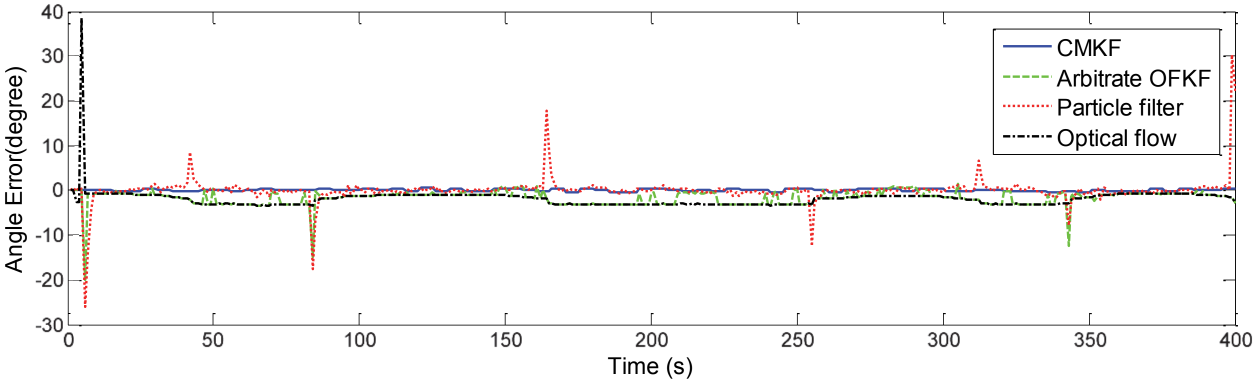


Figure 7 The comparison of tracking error among the methods of optical flow, arbitrate OFKF, particle filter, and CMKF.

Table 2 Methodological comparison of tracking

	Duration of the tracking correctness within 3-sigma bound (%)	Standard deviation of the tracking error under different condition						Bias for residual error during the stable periods
		Ground proof			Noisy			
		Fast	Med	Slow	Fast	Med	Slow	
CMKF	32	0.07	0.08	0.08	0.07	0.08	0.08	0.02
Arbitrate OFKF	3.5	239.36	1.77	1.83	2.50	2.51	5.56	0.41
Particle filter	3.02	0.82	0.42	0.31	1.10	1.25	0.52	0.45
Optical flow	5.01	7.58	0.76	0.60	0.63	0.36	0.74	1.76

the robot, such as 20 s with 90 repetitions in the movement, can match the human's present motion to some motion in the past, and can make much more accurate decisions than the KF implementation.

Figure 11a, b shows the final error comparisons when KF and CMKF are applied to a motion. Note that the mean error decreases by 11.6% when only 20% of the motion of the man is repeated. It increases to 73%, when 90% of the action is repeated. The standard deviation of the error also improves at approximately the same pace.

It can be seen that with more repetition in movement, the error with which the CMKF predicts the position of the human decreases. The error also decreases with time, as there is a larger database to compare the actions with, and there is a high likelihood of the present motion of the human matching with some motion in its past history. The only trade-off is the time required for the algorithm to predict, which is not too much of a drawback as it is of the order of 10e-4s. An optimized equation maybe drawn between the frame rate of the sensor available and the time taken for the CMKF to predict, and the size of the database maybe restricted accordingly (as mentioned in Section 3.6).

5.2. Experimental verification

When curve matching is used with KF, it results in a better performance even when the target temporarily loses sight of the target as shown in Figure 12. The CMKF will, in such a case, try to extrapolate the present motion of the man, using curve matching, and predict the probable position of the human. In the case of the human being outside the field of view, the robot is made to move in accordance with that part of the past motion, which resembles the human's present motion the most.

To check the efficiency of the algorithm, in helping the robots to track a human even when he is out of sight, the position curve of a man was simulated and the loss of track was simulated at various points in the motion. Along a 1000-frame motion of the man, the algorithm was run to predict the position of the man, after a gap of 1 to 50 frames, and the result was compared with the actual position of the human after that many frames. The results showed that the longer the time period for which the human is out of view, the more the prediction error increases. Figure 13a shows a 3D plot of the errors.

Figure 13a shows that for about 20 frames after the robot loses track of its human, the error in its prediction

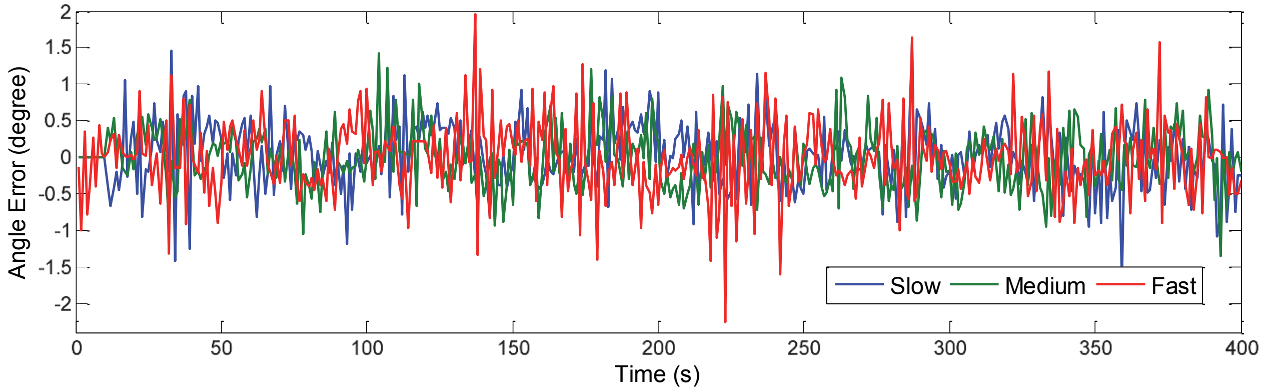


Figure 8 The comparison of tracking error among the multiple trials of different speed target movements.

Table 3 Multiple trials of tracking with different speed using CMFK

Trials of different speed target movement	Mean error	Standard deviation of error	Mean time required per cycle
Slow	-0.0228	0.0809	8.1074e-5
Medium	0.0153	0.0870	8.5299e-5
Fast	0.0129	0.0753	8.5866e-5

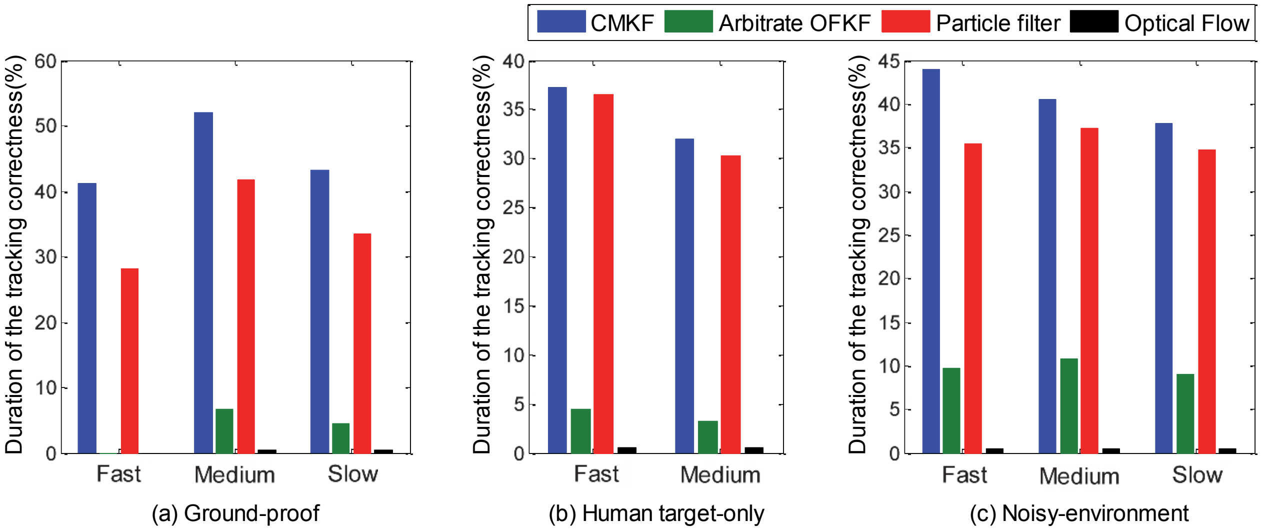


Figure 9 The comparison of percentage duration of tracking correctness for different data conditions: (a) Ground-proof, (b) Human target-only, and (c) Noisy-environment with multiple speed trials of target movement.

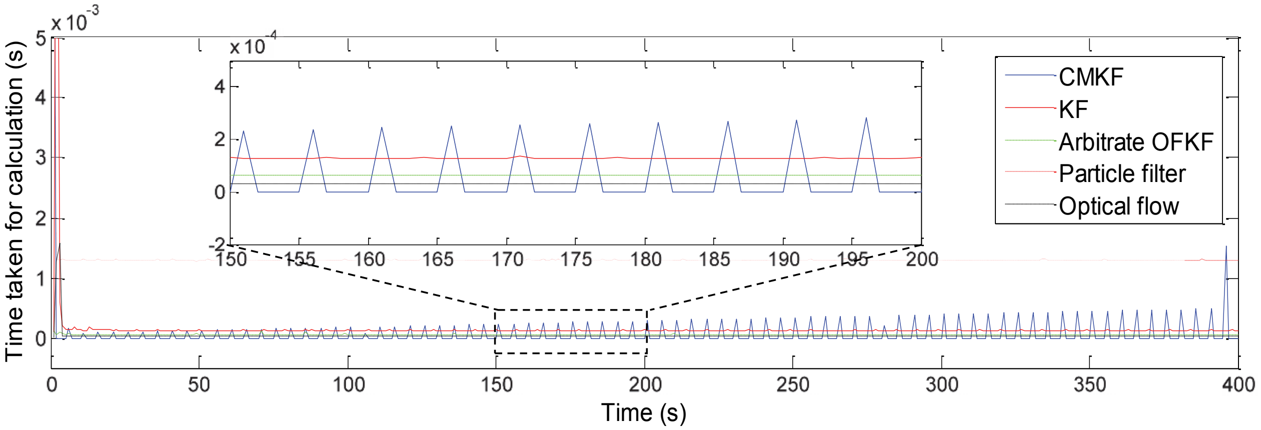


Figure 10 Comparison of time taken for CMKF, KF, arbitrate OFKF, particle filter, and optical flow to work. The camera used has a frame rate of 15 frames/s and the difference in time taken by the algorithms to track is of the order of 10^{-4} s. Thus, it is not expected to slow down the robot when it is tracking the human being.

Table 4 Comparison of average mean time with other methods

	CMKF	KF	Arbitrate OFKF	Particle filter	Optical flow
Average mean time	2.3429e-4	1.6232e-4	6.4351e-5	0.0013	4.1963e-5

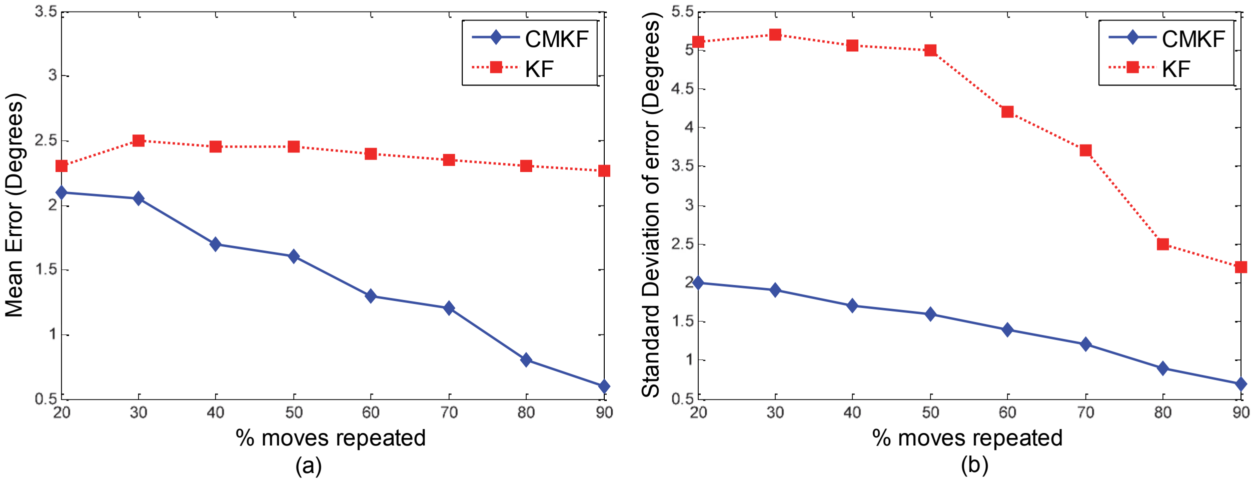


Figure 11 The extent to which the CMKF improves with respect to the extent of repetition in the movement in the human. The two graphs show the variation of the average and the standard deviation of the mean of KF and the CMKF against the percentage of moves that have been matched against a previous history.

based on curve matching and the man's actual motion lies within 20°. As the number of frames in which the human is absent begins to increase, the error in the prediction of the CMA also increases. Figure 13b shows the mean and standard deviation of the error of the predictions made with the assumption that the human is out of sight. It can be seen that for the first few frames after the robot has lost track of the human, it makes accurate guesses about the position of the human, and can be assumed to get the human back into view within a short amount of time.

CMKF will outperform KF in cases where the human has shown a tendency to change its direction while moving, which is normally what happens. KF will not be able to predict the changes in direction of motion of the human, whereas the CMKF will. In situations where the human shows a tendency of smooth motion, the CMKF will give as good a result as the KF because the CMKF will predict a smooth motion matching some portion in the human's past history.

The robot used has a pan-vision of about 45° on each side of its optical axis. If the error is less than 45° at any instant, the robot can get the human back in sight, after making a prediction, but if the error exceeds 45°, then the rotation of the robot will not bring the human into view at all. This happens around frame number 40. Forty frames from the sensor correspond to roughly 3 s (frame rate of the infrared sensor used in the experiment is 15 Hz); so, if the robot does not have the track of the human for more than 3 s, we can assume that it is lost.

When the object goes out of view, for whatever reason, the CMKF will do a much more efficient job of predicting where the human might be, having only the

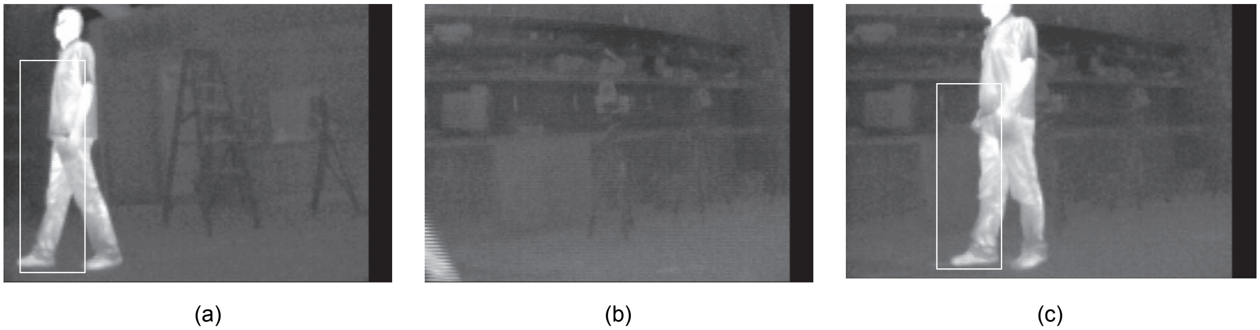
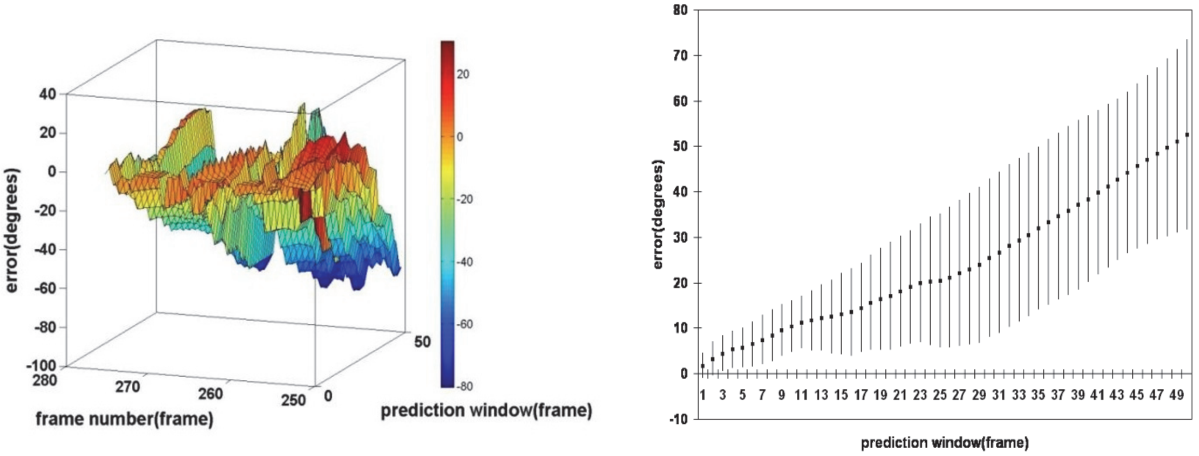


Figure 12 It may happen that due to a burst of speed, the target temporarily gets out of field of view of the sensor. In such cases, the last few values of the target's motion are used to interpolate his motion using curve matching. **(a)** The target picking up speed, while by the time the robot reacts in **(b)** the target is already out of sight. Interpolating the target's motion, the camera swings around and as we see in **(c)** it has the target back in its field of view, whereon it can be tracked normally.



(a) (b)

Figure 13 Prediction accuracy analysis. **(a)** Variation in error of prediction when the human is out of field of the view of the sensors. The frame number axis shows at which frame the camera was simulated to have lost track of the human, and the prediction window axis shows the number of frames for which the algorithm was asked to predict the position of the human after it had lost track, **(b)** mean (represented by the bold dots) and the standard deviation (shown by the extended lines) of the error in prediction as more and more frames pass without the human in view. It can be easily seen that the error of prediction of the human's position keeps increasing as the number of frames within which the object is absent increases. This dataset had a 5.9% repetition of data.

last few values of motion to extrapolate the curve of motion.

6. Conclusion

When curve matching is added to an existing KF model in the integrated manner, it has been shown to improve the performance of KF to an extent as displayed in the experiments of infrared image-based analysis. When the percentage of repeated moves is around 20%, the improvement in mean error is almost 12%.

If the proposed hybrid algorithm is used to track objects or humans with a continuous tendency of repetitive of motion, the mean error improves by 50-70% over a traditional implementation of KF. Also, the algorithm uses exclusively the dataset which is obtained from the human it is tracking-and hence does not need previously stored training datasets. Training datasets, if available, will always boost the performance of the CMKF. This top-down and bottom-up algorithm is thus expected to perform much better than ordinary methods even if the tracked target has a unique but repetitive motion because it has the capability of acquiring the characteristics of the target real time. More accurate CMAs and better human segmentation methods will further enhance the performance of the CMKF on human tracking by fusing more sensors to get help in making the CMKF more accurate. These areas can be probed into for further development of this algorithm.

Acknowledgements
This study was supported in part by the dean's office of the School of Engineering at Virginia Commonwealth University, and NSF ECCS #1054333.

Author details
[1]Department of Electrical and Computer Engineering, Virginia Commonwealth University, Richmond, VA, USA [2]Department of Electrical Engineering and Computer Science, University of California, Berkley, CA, USA

Competing interests
The authors declare that they have no competing interests.

Received: 16 May 2011 Accepted: 3 May 2012 Published: 3 May 2012

References

1. Y Bar-Shalom, X Li, T Kirubarajan, **Estimation With Applications to Tracking and Navigation** (Wiley, New York, 2001)
2. C Nelson, R Polana, Detection and recognition of periodic, non-rigid motion. Int J Comput Vis. **23**, 261–282 (1997)
3. AB Albu, R Bergevin, S Quirion, Generic temporal segmentation of cyclic human motion. Pattern Recognit. **41**, 6–21 (2008)
4. PS Tsai, M Shah, K Keiter, T Kasparis, Cyclic motion detection for motion based recognition. Pattern Recognit. **27**, 1591–1603 (1994)
5. I Mikic, M Trivedi, E Hunter, P Cosman, Human body model acquisition and tracking using voxel data. Int J Comput Vis. **53**, 199–223 (2003)
6. DS Jang, SW Jang, HI Choi, 2D human body tracking with structural Kalman filter. Pattern Recognit. **35**, 2041–2049 (2002)
7. AK Rastogi, BN Chatterji, AK Ray, Design of a real-time tracking system for fast-moving objects. IETE J Res. **43**, 359–369 (1997)
8. N Bellotto, HU HS, Multisensor-based human detection and tracking for mobile service robots. IEEE Trans Syst Man Cybern B Cybernetics. **39**, 167–181 (2009)
9. S Jung, K Wohn, Tracking and motion estimation of the articulated object: a hierarchical Kalman filter approach. Real-Time Imag. **3**, 415–432 (1997)
10. XP Yun, ER Bachmann, Design, implementation, and experimental results of a quaternion-based Kalman filter for human body motion tracking. IEEE Trans Robot. **22**, 1216–1227 (2006)
11. MT Yang, SC Wang, YY Lin, A multimodal fusion system for people detection and tracking. Int J Imag Syst Technol. **15**, 131–142 (2005)
12. L Jetto, S Longhi, G Venturini, Development and experimental validation of an adaptive extended Kalman filter for the localization of mobile robots. IEEE Trans Robot Autom. **15**, 219–229 (1999)
13. C Bauckhage, JK Tsotsos, FE Bunn, Automatic detection of abnormal gait. Image Vis Comput. **27**, 108–115 (2009)

14. XL Xhon, B Bhanu, Integrating face and gait for human recognition at a distance in video. IEEE Trans Syst Man Cybern B Cybernetics. **37**, 1119–1137 (2007)
15. PS Huang, Automatic gait recognition via statistical approaches for extended template features. IEEE Trans Syst Man Cybern B Cybernetics. **31**, 818–824 (2001)
16. L Wang, T Tan, HZ Ning, W Hu, Silhouette analysis-based gait recognition for human identification. IEEE Trans Pattern Anal Mach Intell. **25**, 1505–1518 (2003)
17. M Yeasin, S Chaudhuri, Toward automatic robot programming: learning human skill from visual data. IEEE Trans Syst Man Cybern B Cybernetics. **30**, 180–185 (2000)
18. F Lerasle, G Rives, M Dhome, Tracking of human limbs by multiocular vision. Comput Vis Image Understand. **75**, 229–246 (1999)
19. M Yeasin, S Chaudhuri, Development of an automated image processing system for kinematic analysis of human gait. Real-Time Imag. **6**, 55–67 (2000)
20. P Weckesser, R Dillmann, Modeling unknown environments with a mobile robot. Robot Auton Syst. **23**, 293–300 (1998)
21. R Rosales, S Sclaroff, A framework for heading-guided recognition of human activity. Comput Vis Image Understand. **91**, 335–367 (2003)
22. SG Wu, L Hong, Hand tracking in a natural conversational environment by the interacting multiple model and probabilistic data association (IMM-PDA) algorithm. Pattern Recognit. **38**, 2143–2158 (2005)
23. RH Himberg, Y Motai, Head orientation prediction: delta quaternions versus quaternions. IEEE Trans Syst Man Cybern B Cybernetics. **39**(6), 1382–1392 (2009)
24. C Barrios, Y Motai, Improving estimation of vehicle's trajectory using latest global positioning system with Kalman filtering. IEEE Trans Instrum Meas. **60**(12), 3747–3755 (2011)
25. G Welch, G Bishop, An introduction to the Kalman filter. SIGGRAPH (2001). (course notes)
26. P Maybeck, *Stochastic Models, Estimation and Control* (Academic Press, New York, 1979)
27. J Bohg, Real-times structure from motion using Kalman filtering. PhD Dissertation (2005)
28. P Saintmarc, JS Chen, G Medioni, Adaptive smoothing-a general tool for early vision. IEEE Trans Pattern Anal Mach Intell. **13**, 514–529 (1991)
29. C Knoll, M Alcaniz, V Grau, C Monserrat, MC Juan, Outlining of the prostate using snakes with shape restrictions based on the wavelet transform (Doctoral thesis). Pattern Recognit. **32**, 1767–1781 (1999)
30. J Porrill, S Pollard, Curve matching and stereo calibration. Image Vis Comput. **9**, 45–50 (1991)
31. JY Wang, FS Cohen, 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. **16**, 13–23 (1994)
32. J Bigun, T Bigun, K Nilsson, Recognition by symmetry derivatives and the generalized structure tensor. IEEE Trans Pattern Anal Mach Intell. **26**, 1590–1605 (2004)
33. YP Wang, SL Lee, K Toraichi, Multiscale curvature-based shape representation using B-spline wavelets. IEEE Trans Image Process. **8**, 1586–1592 (1999)
34. DW Paglieroni, GE Ford, EM Tsujimoto, The position-orientation masking approach to parametric search for template matching. IEEE Trans Pattern Anal Mach Intell. **16**, 740–747 (1994)
35. LD Cohen, Auxiliary variables and two-step iterative algorithms in computer vision problems. J Math Imag Vis. **6**, 59–83 (1996)
36. TB Sebastian, BB Kimia, Curves vs. skeletons in object recognition. Signal Process. **85**, 247–263 (2005)
37. BJ Super, Fast correspondence-based system for shape retrieval. Pattern Recognit Lett. **25**, 217–225 (2004)
38. C Gope, N Kehtarnavaz, G Hillman, B Wursig, An affine invariant curve matching method for photo-identification of marine mammals. Pattern Recognit. **38**, 125–132 (2005)
39. A Ross, SC Dass, AK Jain, Fingerprint warping using ridge curve correspondences. IEEE Trans Pattern Anal Mach Intell. **28**, 19–30 (2006)
40. C Orrite, JE Herrero, Shape matching of partially occluded curves invariant under projective transformation. Comput Vis Image Understand. **93**, 34–64 (2004)
41. CF Olson, A general method for geometric feature matching and model extraction. Int J Comput Vis. **45**, 39–54 (2001)
42. FS Cohen, ZH Huang, ZW Yang, Unvariant matching and identification of curves using B-spline curve representation. IEEE Trans Image Process. **4**, 1–17 (1995)
43. ZH Huang, FS Cohen, Affine-invariant B-spline moments for curve matching. IEEE Trans Image Process. **5**, 1473–1480 (1996)
44. E Kishon, T Hastie, H Wolfson, 3-D curve matching using splines. J Robot Syst. **8**, 723–743 (1991)
45. Y Avrithis, Y Xirouhakis, S Kollias, Affine-invariant curve normalization for object shape representation, classification, and retrieval. Mach Vis Appl. **13**, 80–94 (2001)
46. YH Gu, T Tjahjadi, Coarse-to-fine planar object identification using invariant curve features and B-spline modeling. Pattern Recognit. **33**, 1411–1422 (2000)
47. B KamgarParsi, Matching sets of 3D line segments with application to polygonal arc matching. IEEE Trans Pattern Anal Mach Intell. **19**, 1090–1099 (1997)
48. PL Rosin, Representing curves at their natural scales. Pattern Recognit. **25**, 1315–1325 (1992)
49. Y Shan, ZY Zhang, New measurements and corner-guidance for curve matching with probabilistic relaxation. Int J Comput Vis. **46**, 157–171 (2002)
50. JS Marques, A fuzzy algorithm for curve and surface alignment. Pattern Recognit Lett. **19**, 797–803 (1998)
51. M Frenkel, R Basri, Curve matching using the fast marching method, in *Proceedings of the Energy Minimization Methods in Computer Vision and Pattern Recognition*. **2683**, 35–51 (2003). Portugal
52. HJ Wolfson, On curve matching. IEEE Trans Pattern Anal Mach Intell. **12**, 483–489 (1990)

doi:10.1186/1687-6180-2012-99
Cite this article as: Lee *et al.*: Human tracking with an infrared camera using curve matching framework. *EURASIP Journal on Advances in Signal Processing* 2012 **2012**:99.