EFFICIENT OBJECT TRACKING USING CONTROL-BASED OBSERVER DESIGN

Wei Qu and Dan Schonfeld

ECE Department, University of Illinois
Chicago, IL, USA, 60607 Email: {wqu, ds}@ece.uic.edu

ABSTRACT

Kernel-based tracking approaches have proven to be more efficient in computation compared to other tracking approaches such as particle filtering. However, existing kernel-based tracking approaches suffer from the well-known "singularity" problem. In this paper, we propose a novel object tracking framework to handle this problem by using a control-based observer design. Specifically, we formulate object tracking as a recursive inverse problem, thus unifying several approaches to video tracking, including kernel-based tracking, into a consistent theoretical framework. Moreover, we interpret the inverse equation as a measurement process and supplement it by introducing state dynamics as a constraint. The augmented recursive inverse equation forms a state-space model, which is solved by using a control-based optimal observer. By exploiting observability theory from control engineering, we extend the current approach to kernel-based tracking and provide explicit criteria for kernel design and dynamics evaluation. The tracking performance of our approach has been demonstrated on both synthetic and real-world video data.

1. INTRODUCTION

Kernel-based tracking [1] has demonstrated its promising performance compared to other tracking approaches such as optical flow and particle filtering due to its much lower computational cost. However, in video sequences containing "complex" scenes such as scale changes, fast motion, or occlusion in cluttered scenes, the basic kernel-based tracking approach suffers from the well-known "singularity" problem in which the tracked object's state cannot be uniquely determined from the observations. This problem usually makes the tracker unstable and often results in complete failure.

Kernel-based tracking is achieved by first using a spatially-weighted color histogram as the object model and then searching its best matches by optimization schemes such as mean shift [1]. Earlier efforts to handle the "singularity" problem have focused on both aspects. Collins [2] proposed to use a scale kernel in addition to the regular spatial kernel presented in [1] in order to recover object scale changes. Multiple spatially distributed kernels were used to increase the tracker's sensitivity by Hager et al. in [3]. Central to their development is a Newton-style optimization method which has been shown to be more efficient than mean shift. This approach was further developed by Fan et al. in [4] who used multiple kernels to enhance the "kernel observability" for articulated objects. They viewed the linear formulation as an observation equation and expanded it by imposing constraints on the observations that satisfy the characteristics of articulated objects. Despite the progress in the use of multiple kernels for object tracking, the underlying principles of kernel design required to solve the "singularity" problem remains an open problem.

Implicit in all approaches to object tracking is the solution to an inverse problem: determine the state of the tracked object from the observations. The theory of inverse problems [5] has been applied in many applications. Many earlier efforts in object tracking relied on control theory to provide a solution. In particular, a state space model representing the state process and measurement process was hypothesized based on physical and statistical models. An optimal observer was used to estimate the state parameters using techniques such as the Kalman filter [6]. This approach can be viewed as a maximum-a-posteriori (MAP) estimate of the conditional density function of the state given the observations. For example, a kernel-based target localization integrated with a Kalman filter has been presented by Comaniciu et al. in [1], where the system and measurement matrices are assumed to be known. More recently, particle filters were used to extend Kalman filters for object tracking by simultaneously tracking multiple samples of the conditional density function.

Unlike the classical formulation of the observation process, a linear observation process is derived from the kernel equation in [3] and [4]. Implicit in their presentation, although not explicitly stated, is the solution of the linear equation as an inverse problem. The tracking parameters are estimated using a recursive optimization of a cost function in [3]. This approach is extended by relying on regularization theory to provide a solution to the constrained linear equation for articulated objects in [4].

In this paper, we extend the kernel-based approaches by formulating object tracking as a recursive inverse problem, thus providing a unified mathematical framework for a class of methods used for object tracking. Similar to the approach presented in [4], we view the linear equation as a measurement process. However, in our approach, we introduce the state dynamics to augment the linear equation and form a

state-space model. A Kalman filter-based observer is proposed to estimate the state parameters, where the observation and state transform matrices are dynamically estimated by the observability analysis. System observability is analyzed by the theory of linear control and used as an extended criterion for both kernel design and dynamics evaluation.

2. OBJECT TRACKING AS AN INVERSE PROBLEM

An inverse problem for object tracking can be defined as follows:

Let x_t represent the object's state (position, velocity, shape and so on) at time t. The observation (image resource such as color, edge and so on) at time t is described by the equation

$$y_t = g_t(x_t),\ g_t : \mathbb{X} \rightarrow \mathbb{Y} \tag{1}$$

where $\mathbb{X}$ and $\mathbb{Y}$ are Banach spaces and g_t is a linear/nonlinear operator. The inverse problem is to determine the state x_t from observation y_t, namely, the inverse of operator g_t^{-1}. The "singularity" problem discussed in the introduction is also called "ill-posed" problem in the theory of inverse problem [5] where if g_t^{-1} does not exist, the solution of equation (1) can not be uniquely determined.

If g_t is nonlinear, it is usually hard to get an analytic solution of (1). To see the predominance of object's state, we can approximate g_t by a linear operator. The linearization can be achieved by dropping higher order terms of Taylor series.

$$g_t(x_t) = g_t(x_0) + \frac{1}{2} g_t'(x_0)(x_t - x_0) \tag{2}$$

Thus we have the linear observation equation

$$\tilde{y}_t = C_t \tilde{x}_t \tag{3}$$

where $\tilde{y}_t = y_t - g_t(x_0)$, $\tilde{x}_t = x_t - x_0$, and $C_t = \frac{1}{2} g_t'(x_0)$.

Since observation y_t is noisy due to the measure error, thus the solution of (1) is also equivalent to the least-square-error optimization method.

$$\hat{x}_t = \arg\min \parallel y_t - g_t(x_t) \parallel^2 \tag{4}$$

A method deriving a solution to the kernel-based tracking problem has been introduced in [3] and reformulated in [4]. We extend this approach to represent any object tracking problem as follows. Consider a cost function for object tracking,

$$\rho[q_t(x_0), p_t(x_t)] = \parallel \sqrt{q_t(x_0)} - \sqrt{p_t(x_t)} \parallel^2 \tag{5}$$

where $\parallel \cdot \parallel$ is the Matusita metric, $q_t(x_0)$ is object's prior model, $p_t(x_t)$ is a function of candidate object region. For example, $q(\cdot)$ and $p(\cdot)$ can be feature histogram, template representation, or probability density etc. Let $y_t = \sqrt{q_t(x_0)}$, $g_t = \sqrt{p_t(x_t)}$, it can be proved that the optimal solution of cost function (5) is the same as the solution of the linear equation $\tilde{y}_t = C_t \tilde{x}_t$, where $\tilde{y}_t = \sqrt{q_t(x_0)} - \sqrt{p_t(x_0)}$, the new state is $\tilde{x}_t = x_t - x_0$, and $C_t = \frac{1}{2}(p_t(x_0))^{-\frac{1}{2}} p_t'(x_0)$.

When limiting the state as object's center $x = c$, and using a kernel-based color histogram for $q(\cdot)$ and $p(\cdot)$, it becomes the Newton-style approach with SSD presented in [3]. In this case, $q(c_0) = U^T K(c_0)$, $p(c) = U^T K(c)$, and $C = \frac{1}{2}\text{diag}[p(c_0)]^{-\frac{1}{2}} U^T J_K(c_0)(c - c_0)$ where U is a sifting matrix indicating which object pixel belong to which bins, K is a vector of the kernel function, J_K is the Jacobian matrix of kernel vector K, and diag$[p]$ represents the matrix with p on its diagonal.

2.1. Improving the tracking performance by enhancing the rank of observation transform matrix C_t

Solving the inverse problem of equation (3) is not trivial. The dimensionality of state and observation is usually different and thus the matrix C_t is not square. This can be solved by *singular value decomposition* [5] which gives a psuedo-inverse of C. The primary difficulty with "ill-posed" problem is that the state is undermined due to small (or zero) singular values of C_t. In other words, if the rank(C_t)$<$ n where n is the dimensionality of state $\tilde{x}$, then C_t^{-1} does not existed. To improve the tracking performance, the earlier efforts of kernel-based approaches [2], [3], [4] can be viewed as enhancing the rank(C_t) by the following two ways:

(1) **Using multiple kernels to enhance the rank(C_t)**

It has been shown in [2], [3] that multiple kernels can improve the tracking performance. But it's not clear why multiple kernels work better than only one in theory and how multiple kernels should be designed. Fan et al. [4] investigated these issues in the context of articulated object. By formulating as an inverse problems, now we can answer these questions more explicitly: M kernels can construct M observation equations, $y_t^m = C_t^m x_t, m = 1, \ldots, M$. By different ways to combine them, rank(C) has potential to be increased. For example, by stacking as [3], $C_t = [C_t^1, \ldots, C_t^M]^T$, so rank($C_t$)$\geq$rank($C_t^m$). Thus, the principle of kernel design is that the additional kernel should help to enhance the rank(C_t).

(2) **Tikhonov regularization to enhance the rank(C_t)**

A kernel-based method using joint state representation and a length constraint among states has been presented in [4] for articulated object tracking. We extend this approach for any object tracking with constraints by using the well-known *Tikhonov regularization* [5]. To cope with the "ill-posed" problem, additional prior information of the state may allow us to select the solution from several feasible estimates. As mentioned, solving the inverse problem can also be viewed as minimizing a cost function such as (4). Tikhonov regularization instead introduces other constraints into the cost function, for example,

$$\hat{x} = \arg\min\{\parallel y_t - C x_t \parallel^2 + \lambda \parallel b - G x_t \parallel^2\} \tag{6}$$

where the regularization parameter $\lambda > 0$.

By using generalized singular value decomposition, it can be shown that (6) has the same solution with linear equation [5],

$$C_t^T y_t + \lambda G^T b = (C_t^T C_t + \lambda G^T G) x_t \quad (7)$$

Thus the new observation matrix $\tilde{C}_t = (C_t^T C_t + \lambda G^T G)$. By selecting λ, it is expected rank($\tilde{C}_t$)$\geq$rank(C_t). Therefore, Tikhonov regularization has the potential to improve the tracking performance.

As we can see, all the analyzed kernel-based approaches [2], [3], [4] did not exploit the state dynamics to cope with the "ill-posed" problem. Although multiple kernels have shown the efficiency to improve the tracking performance, this solution is not good enough. In the experiments, we found that in case of fast motions (in other words, the low frame rate video) or occlusions, the kernel-based approaches usually still suffer from the "ill-posed" problem and can not track object robustly. This further inspired us to formulate object tracking as a recursive inverse problem which can include object's state dynamics to solve the "ill-posed" problem better.

3. CONTROL-BASED OBSERVER DESIGN FOR EFFICIENT OBJECT TRACKING

Video tracking can be formulated by a recursive linear inverse problem when using the state dynamics. Consider the stochastic system represented by the state and observation equations,

$$x_{t+1} = A_t x_t + w_t \quad (8)$$

$$y_t = C_t x_t + v_t \quad (9)$$

where the system is corrupted by an additive random noise signal w; and the observation is corrupted by noise v.

When matrix A_t and C_t are known and noise term w_t and v_t are both Gaussian, this system can be solved by Kalman filter [7]. A method of Kalman filter based tracking approach was presented in [6]. Comaniciu used this approach with a kernel-based target model in [1]. In both of them, the transform matrices are assumed to be known and fixed. However, these conditions are not satisfied for practical video tracking problems where the state dynamics is usually unknown and may be time-variant. Different motion estimation techniques and scene-based prior knowledge can be used to roughly approximate the dynamics. For example, we can assume A is an identity matrix according to the inertia. To select the best motion estimate for handling the "ill-posed" problem, we need a criterion to evaluate which one can most increase the possibility of uniquely "observing" the state? This inspired us to introduce the *observability theory* in control engineering [7]. It has been proved that the observability of a linear system describe by (8) and (9) can be determined as follows [7]:

Observability Theorem: *A system is observable if and only if its observability matrix $\mathcal{O}_t$ has full rank, i.e., rank($\mathcal{O}_t$) = n, where $\mathcal{O}_t = [C_t,\ C_t A_t,\ \ldots,\ C_t A_t^{n-1}]^T \in \mathbb{R}^{pn \times n}$, $A_t \in \mathbb{R}^{n \times n}$ and $C_t \in \mathbb{R}^{p \times n}$.*

This theorem is consistent with our earlier analysis for non-recursive inverse problem where the observability matrix $\mathcal{O}_t$ degrades to matrix C_t since there is no state equation at all in that case. It is obvious that by exploiting state dynamics, rank($\mathcal{O}_t$)$\geq$rank(C_t), namely the observability of recursive system is not less than the system of (3). Thus, the recursive system can cope with the "ill-posed" problem better than the approach without state dynamics. Guided by the observability theorem, we show a paradigm of solving the recursive inverse problem by a control-based observer.

3.1. A Paradigm of Control-Based Observer

Different from the regular Kalman filter tracking approaches [1], [6], we use prior knowledge and/or different motion estimation methods to estimate the dynamics, $\{A_t^1, \ldots, A_t^J\}$. At each time, we select the optimal A_t^j which can make rank($\mathcal{O}_t$) highest. It can make the observer have more possibilities to determine the state uniquely. If several ones have the same highest rank, we choose the one most similar to the previous time step. Then the selected A_t^j can construct a Kalman-Bucy filter and the estimate of state can be given by [7, p.480-495],

$$\hat{x_t} = [I - L_t C_t] A_t x_{t-1} + L_t y_t \quad (10)$$

where I is the identity matrix, L_t is the filter gain matrix. We briefly refer to this approach as TUCO (Tracking Using Control-based Observer).

4. EXPERIMENTAL RESULTS

The performance of the proposed TUCO has been demonstrated on both synthetic and real-world video sequences. They are captured by a resolution of 320×240 pixels with a frame rate of 30fps. In all the experiments, we use a multiple kernel-based color histogram similar to MKT-SSD [3] for better comparison, which has 10 bins for RGB channels respectively.

The synthetic video has a book moving according to predefined state dynamics in a clutter scene. The changing background prevents background subtraction from easily object tracking. We have tested our approach and MKT-SSD [3] on sequences with original frame rate of 30fps and a lower frame rate of 10fps where object's motion becomes much faster. Fig. 1 presents the tracking trajectories of object's center. As we can see, for the 30fps-sequence, although MKT-SSD can achieve comparably tracking results as our approach for most part of the sequence, it fails after frame number 332 due to the "ill-posed" problem. For the 10fps-sequence, MKT-SSD suffers from the fast motion and loses the object after frame number 90. However, due to exploiting the state dynamics and coping the "ill-posed" problem, our approach achieves more robust performance in both cases. Fig. 2 illustrates the

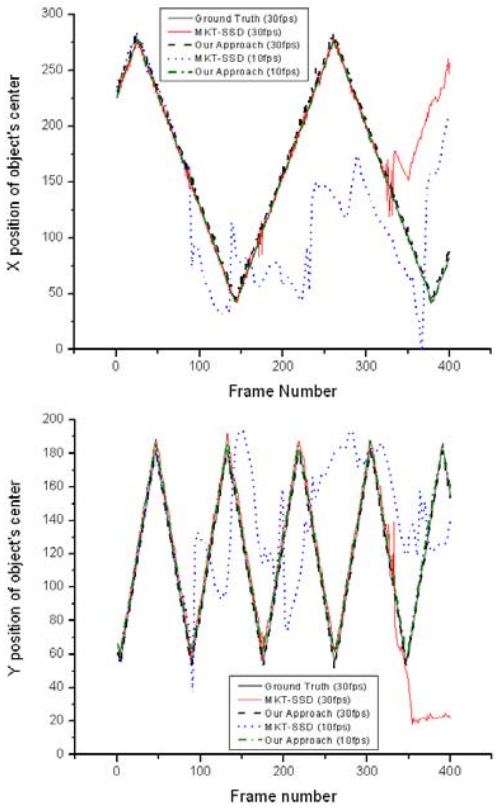

Fig. 1. The tracking trajectories of object's center using our TUCO and MKT-SSD [3] for the synthetic sequence with different frame rate.

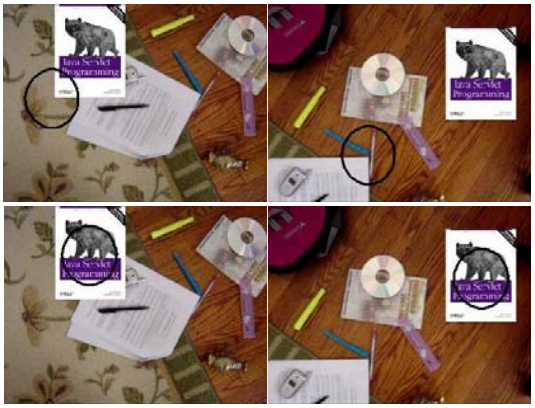

Fig. 2. The tracking results using MKT-SSD [3] (1st row) and our TUCO (2nd row) respectively.

Fig. 3. Tracking results using KF [1], [6] (black), MKT-SSD [3] (green) and our TUCO (white) for multiple object tracking in a crowded scene. The first image is the initial frame.

tracking results for the 10fps-sequence where 1st row was implemented by MKT-SSD and 2nd row used our approach.

We further compare the performance of our TUCO to MKF-SSD [3] and the regular Kalman filter tracking approach (KF) [1], [6] on a real-world video sequence. It has a crowded scene presenting various motions and different occlusions. We use two independent trackers for two pedestrians with remarkable color features. Two matrices A_1 and A_2 are used to estimate the state dynamics where A_1 is assumed to be an identity matrix and A_2 is estimated by background subtraction and object matching. The tracking results are illustrated in Fig. 3 where we use ellipses of different colors to show the results of different approaches. As we can see, KF (black) is helpful to handle the fast motion. But this approach badly suffers from the background clutter and can not observe the change of object's scale. MKT-SSD can handle object's scale change and the tracking results are more accurate when there is no occlusion and fast motion. However, both of them suffer from the "ill-posed" problem and fail to track object robustly and consistently. Our approach can achieve more robust tracking results handling both partial occlusion, fast motion in the crowded scene.

5. ACKNOWLEDGEMENT

We would like to thank Mannesh Dewan for sharing part of the code implementing MTK-SSD [3].

6. REFERENCES

[1] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *PAMI*, vol. 25, no. 5, pp. 564–577, 2003.

[2] R. T. Collins, "Mean-shift blob tracking through scale space," in *CVPR*, 2003, vol. 2, pp. 234–240.

[3] G. D. Hager, M. Dewan, and C. V. Stewart, "Multiple kernel tracking with *SSD*," in *CVPR*, 2004, vol. 1, pp. 790–97.

[4] Z. Fan, Y. Wu, and M. Yang, "Multiple collaborative kernel tracking," in *CVPR*, 2005, vol. 2, pp. 502–509.

[5] A. G. Ramm, *Inverse Problem*, Springer, 2005.

[6] Y. Bar-Shalom and T. Fortmann, *Tracking and Data Association*, Academic Press, 1988.

[7] Ken Dutton, Steve Thompson, and Bill Barraclough, *The Art of Control Engineering*, Prentice Hall, 1997.