

ROBUST KERNEL-BASED TRACKING USING OPTIMAL CONTROL

Wei Qu and Dan Schonfeld

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

ABSTRACT

Although more efficient in computation compared to other tracking approaches such as particle filtering, the kernel-based tracking suffers from the “singularity” problem which makes the tracking unstable and even completely fail. In this paper, we propose a novel framework to handle this problem by enhancing the tracker’s observability. In particular, we formulate object tracking as an inverse problem, thus unifying the existing kernel-based tracking approaches into a consistent theoretical framework. By exploiting the observability theory, we explicitly give the criterion for kernel design and constraint selection. Moreover, we extend the kernel-based approach by including the state dynamics and thus form a state-space model. The use of observability theory is also extended for dynamics estimation and evaluation. We rely on an optimal observer for state estimation as a solution to video tracking. The performance of the proposed approach has been demonstrated on both synthetic and real-world video data and compared to other kernel-based tracking approaches.

Index Terms— Tracking, optimal control, inverse problem, singularity problem

1. INTRODUCTION

Video tracking has received much attention due to its wide applications. Kernel-based tracking [1] has demonstrated its promising performance compared to other tracking approaches because of its much lower computational cost. The efficient tracking is achieved by modeling the tracking object with a spatially weighted feature histogram and searching the best match with optimization techniques such as mean-shift [1]. However, in video sequences containing “complex” scenes such as fast motion, and occlusion in clutter scenes, the basic kernel-based tracking approach usually suffers from the well-known “singularity” problem in which the tracked object’s state can not be uniquely determined from the observations.

Earlier efforts to handle the “singularity” problem were mainly by using multiple kernels. 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 the linearization of the kernel tracking equation

and its subsequent solution using 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 principle of kernel design required to solve the “singularity” problem remains an open problem. Kalman filter has been used in tracking for many years [5]. However, how to determine the state and observation transform functions are not trivial and still limit its performance for video tracking. For example, the state and measurement processes are usually assumed to be known based on physical and statistical models. This is reasonable for some specific applications such as aircraft tracking. But for generic video tracking, this assumption is often unsatisfied. Kernel-based approach has potential to be integrated with Kalman filter for better tracking performance. Comaniciu et al. [1] presented an approach combining the kernel-based target localization with a Kalman filter by fitting the similarity surface with a scaled Gaussian. The state and measurement matrices were still assumed to be constant.

This paper extends the existing kernel-based approaches [1], [2], [3], [4] into a more generic framework by formulating object tracking as a recursive inverse problem. By exploiting the observability theorem, the criterion of kernel design is explicitly presented. We view the linear equation as an observation process. Furthermore, we exploit the state dynamics to form a state-space model. The state parameters are estimated by a proposed Kalman filter-based observer. Unlike the classic formulation of the Kalman filter [1], [5] where the state and observation processes are assumed to be known, in our approach, the observation equation is derived from kernel-based optimization and the state dynamics is dynamically estimated and selected by observability analysis.

2. AN INVERSE PROBLEM FORMULATION FOR VIDEO TRACKING

Inspired by the kernel-based tracking approaches in [1], [3], we formulate the object tracking as an inverse problem [6]:

Define \mathbf{x}_t as an object's state at time t . The observation at time t is represented by the equation $\mathbf{y}_t = f_t(\mathbf{x}_t)$, where f_t is a linear/nonlinear operator. The inverse problem of object tracking is to determine the state \mathbf{x}_t from observation \mathbf{y}_t , namely, the inverse of operator f_t^{-1} . The "singularity" problem discussed in the introduction is also called "ill-posed" problem in the theory of inverse problem [6].

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

$$f_t(\mathbf{x}_t) = f_t(\mathbf{x}_0) + \frac{1}{2}f_t'(\mathbf{x}_0)(\mathbf{x}_t - \mathbf{x}_0). \quad (1)$$

Thus we have a linear observation equation for tracking problem

$$\tilde{\mathbf{y}}_t = \mathbf{C}_t \tilde{\mathbf{x}}_t, \quad (2)$$

where $\tilde{\mathbf{y}}_t = \mathbf{y}_t - f_t(\mathbf{x}_0)$, $\tilde{\mathbf{x}}_t = \mathbf{x}_t - \mathbf{x}_0$, and $\mathbf{C}_t = \frac{1}{2}f_t'(\mathbf{x}_0)$. The solution of (2) is also equivalent to the optimization method

$$\hat{\tilde{\mathbf{x}}}_t = \arg \min \rho(\tilde{\mathbf{y}}, \mathbf{C}_t \tilde{\mathbf{x}}_t), \quad (3)$$

where different metrics can be employed for the cost function $\rho(\cdot)$ such as the Bhattacharyya distance [1], and the Matusita distance [3].

Hager et al. proposed a solution to the kernel-based tracking problem in [3]. Fan et al. [4] reformulated this method for articulated object tracking. We extend this approach to represent any object tracking problem as follows: Consider a cost function for object tracking

$$\rho[q_t(\mathbf{x}_0), p_t(\mathbf{x}_t)] = \|\sqrt{q_t(\mathbf{x}_0)} - \sqrt{p_t(\mathbf{x}_t)}\|^2, \quad (4)$$

where $\|\cdot\|$ is the Matusita metric, $q_t(\mathbf{x}_0)$ is object's prior model, $p_t(\mathbf{x}_t)$ is a function of candidate object region. For example, $q(\cdot)$ and $p(\cdot)$ can be feature histogram, template representation, probability density etc.. Let $\mathbf{y}_t = \sqrt{q_t(\mathbf{x}_0)}$, $f_t = \sqrt{p_t(\mathbf{x}_t)}$, it can be proved that the optimal solution of cost function (4) is the same as the solution of the linear equation $\tilde{\mathbf{y}}_t = \mathbf{C}_t \tilde{\mathbf{x}}_t$, where $\tilde{\mathbf{y}}_t = \sqrt{q_t(\mathbf{x}_0)} - \sqrt{p_t(\mathbf{x}_0)}$, the new state is $\tilde{\mathbf{x}}_t = \mathbf{x}_t - \mathbf{x}_0$, and $\mathbf{C}_t = \frac{1}{2}(p_t(\mathbf{x}_0))^{-\frac{1}{2}}p_t'(\mathbf{x}_0)$. 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 in [3]. In this case, $q(c_0) = \mathbf{U}^T \mathbf{K}(c_0)$, $p(c) = \mathbf{U}^T \mathbf{K}(c)$, and $\mathbf{C} = \frac{1}{2} \text{diag}[p(c_0)]^{-\frac{1}{2}} \mathbf{U}^T \mathbf{J}_{\mathbf{K}}(c_0)(c - c_0)$ where \mathbf{U} is a sifting matrix indicating which object pixels belong to which bins, \mathbf{K} is a vector of the kernel function, $\mathbf{J}_{\mathbf{K}}$ is the Jacobian matrix of kernel vector \mathbf{K} , and $\text{diag}[p]$ represents the matrix with p on its diagonal.

3. ROBUST KERNEL-BASED TRACKING BY OBSERVABILITY ENHANCEMENT

How to solve the inverse problem of (2) or (3) is not trivial. The dimensionality of state and observation is usually different and thus the matrix \mathbf{C}_t is not square. This can be solved

by *singular value decomposition* [6] which gives a psuedo-inverse of \mathbf{C} . The primary difficulty with "singularity" problem is that the state is undermined due to small (or zero) singular values of \mathbf{C}_t . To handle the "singularity" problem, we can easily define the system observability as follows:

Observability Theorem I: *A system is observable if and only if its observability matrix \mathbf{C}_t has full rank, i.e., $\text{rank}(\mathbf{C}_t) = n$, where $\mathbf{C}_t \in \mathbb{R}^{p \times n}$.*

With this observability definition, the earlier efforts of kernel-based approaches [2], [3], [4] can somehow be viewed as enhancing the tracker's observability by the following two ways: using multiple kernels and by Tikhonov regularization. It has been shown in [2], [3] that using multiple kernels can improve the tracking performance. However, it's not clear, in principle, why multiple kernels work better than only a single one and how multiple kernels should be designed. Fan et al. [4] initially investigated these issues in the context of articulated object. By formulating video tracking as an inverse problem, we can answer these questions more explicitly and generically: M kernels can construct M observation equations, $\mathbf{y}_t^m = \mathbf{C}_t^m \mathbf{x}_t$, $m = 1, \dots, M$. By different ways to combine them, $\text{rank}(\mathbf{C})$ has potential to be increased. For example, by stacking as [3], $\mathbf{C}_t = [\mathbf{C}_t^1, \dots, \mathbf{C}_t^M]^T$. Thus, $\text{rank}(\mathbf{C}_t) \geq \text{rank}(\mathbf{C}_t^m)$. Therefore, the principle of kernel design is that the additional kernel should help to enhance the $\text{rank}(\mathbf{C}_t)$.

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* [6]. To cope with the "singularity" 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 (3). Tikhonov regularization instead introduces other constraints into the cost function, for example,

$$\hat{\tilde{\mathbf{x}}} = \arg \min \{\|\tilde{\mathbf{y}}_t - \mathbf{C} \tilde{\mathbf{x}}_t\|^2 + \lambda \|\mathbf{b} - \mathbf{G} \tilde{\mathbf{x}}_t\|^2\}, \quad (5)$$

where the regularization parameter $\lambda > 0$. By using generalized singular value decomposition, it can be shown that (5) has the same solution with the linear equation [6]

$$\mathbf{C}_t^T \tilde{\mathbf{y}}_t + \lambda \mathbf{G}^T \mathbf{b} = (\mathbf{C}_t^T \mathbf{C}_t + \lambda \mathbf{G}^T \mathbf{G}) \tilde{\mathbf{x}}_t. \quad (6)$$

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

So far, all the analyzed kernel-based approaches did not fully exploit abundant temporal information among video sequence but only assume the searching start point is the object's location in the previous frame. In the experiments, we found that these approaches still suffered from the "singularity" problem and could not track object robustly especially in

case of fast motions or occlusions where object’s motion information is supposed to be very helpful to improve the performance. This further inspired us to include object’s state dynamics to solve the “singularity” problem better.

Consider the stochastic system represented by the state and observation equations,

$$\mathbf{x}_{t+1} = \mathbf{A}_t \mathbf{x}_t + \mathbf{w}_t \quad (7)$$

$$\mathbf{y}_t = \mathbf{C}_t \mathbf{x}_t + \mathbf{v}_t, \quad (8)$$

where the system is corrupted by an additive random noise signal \mathbf{w} ; and the observation is corrupted by noise \mathbf{v} . When matrix \mathbf{A}_t and \mathbf{C}_t are known and noise term \mathbf{w}_t and \mathbf{v}_t are both Gaussian, this system can be solved by Kalman filter [7]. For example, a Kalman filtering method for target tracking was presented in [5]. However, in the context of practical video tracking, both the state dynamics \mathbf{A}_t and the observation matrix \mathbf{C}_t are usually unknown. Moreover, the noise terms may be not Gaussian. All these conditions limit the performance of Kalman filter for video tracking. Comaniciu et al. showed a promising direction of combining the kernel-based target localization with Kalman filter by first fitting the similarity surface with a Gaussian density and then predicting this density with Kalman filter in [1]. However, the transform matrices were assumed to be constant. We extend this idea but integrate the kernel-based approach with Kalman filter in a different way.

In the system described by (7) and (8), the observation equation can be got from the kernel-based method as we derive in the earlier section. The state equation can be estimated by different motion estimation techniques and/or scene-based prior knowledge. For example, we can always assume A is an identity matrix according to object’s inertia without loss of generality. This introduces another problem: how should we select the best motion estimate if we have several ones through different approaches. There are different criteria to define the “best”. To cope with the “singularity” problem, the best we want is the one that can most increase the possibility of uniquely “observing” the state. We further borrow the idea of *observability theory* from control engineering [7] to answer this question. It has been proved that the observability of a linear system describe by (7) and (8) can be determined as follows [7]:

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

When not exploiting the state equation, the above theorem is consistent with **Observability Theorem I** for non-recursive inverse problem where the observability matrix \mathcal{O}_t degrades to matrix \mathbf{C}_t . It is clear that by exploiting state dynamics, $\text{rank}(\mathcal{O}_t) \geq \text{rank}(\mathbf{C}_t)$, namely, the observability of recursive system is not less than the observation equation itself. Thus, the recursive system can cope with the “singularity”

problem better than non-recursive system without state dynamics.

Guided by the **Observability Theorem II**, we show an example of solving the recursive inverse problem as follows: Suppose we have different estimates of the state transform matrix using prior knowledge and different motion estimation methods, $\{\mathbf{A}_t^1, \dots, \mathbf{A}_t^j\}$. At each time, we select the optimal \mathbf{A}_t^j which can make $\text{rank}(\mathcal{O}_t)$ highest. It can make the tracker have more possibilities to determine the state uniquely. Then the selected optimal \mathbf{A}_t^j can construct a Kalman-Bucy filter and the estimate of state is [7]

$$\hat{\mathbf{x}}_t = [\mathbf{I} - \mathbf{L}_t \mathbf{C}_t] \mathbf{A}_t \mathbf{x}_{t-1} + \mathbf{L}_t \mathbf{y}_t, \quad (9)$$

where \mathbf{I} is the identity matrix, \mathbf{L}_t is the filter gain matrix.

4. EXPERIMENTAL RESULTS

We have tested the proposed approach on both synthetic and real-world video sequences, which have a resolution of 320×240 pixels with a frame rate of 30fps. For better comparisons with existing approaches, we used a multiple kernel-based color histogram, which has 10 bins for RGB channels respectively.

The synthetic video `BOOK` has a book moving in a changing clutter environment. Object’s state dynamics is predefined. We have compared the performance of our approach and MKT-SSD [3] on sequences with original frame rate of 30fps and a lower frame rate of 10fps where the motion becomes much faster. The tracking trajectories of object’s center are shown in Fig. 1. It can be seen that MKT-SSD suffered from the “singularity” problem and could not produce satisfactory tracking results. It failed to track object especially when the fast motion presented in the 10fps sequence. However, due to exploiting the state dynamics and solving the “singularity” problem with control-based observer design, the proposed approach achieved robust tracking performance in both cases. Fig. 2 shows some tracking frames for the 10fps-sequence where 1st row was implemented by MKT-SSD and 2nd row used our approach.

We further compared the performance of the proposed approach with MKF-SSD [3] and the regular Kalman filter tracking approach (KF) [1], [5] on a real-world video sequence. It has a crowded scene presenting various motions and occasional different occlusions. We used two independent trackers for two pedestrians with remarkable color features. Two matrices \mathbf{A}_1 and \mathbf{A}_2 were used to estimate the state dynamics where \mathbf{A}_1 was assumed to be an identity matrix and \mathbf{A}_2 was estimated by background subtraction and object matching. The tracking results are illustrated in Fig. 3 where the top-left image is the initial frame. In each of the following frames, we use ellipse of different colors to show the results of different approaches. As we can see, the regular Kalman filter approach (black) is helpful to handle the fast motion. But this approach badly suffered from the background clutter

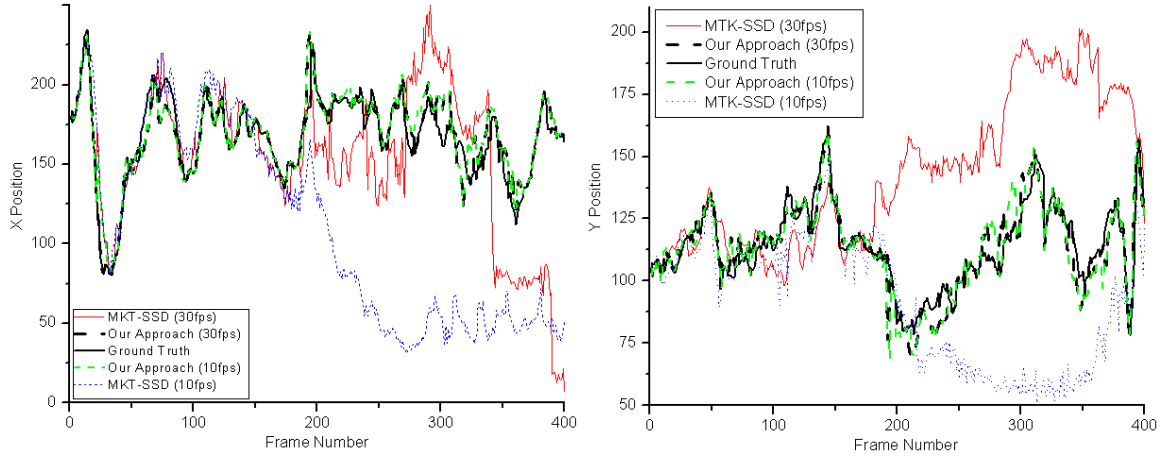


Fig. 1. The tracking trajectories of object's center using the proposed approach and MKT-SSD [3] for the synthetic sequence with different frame rate.

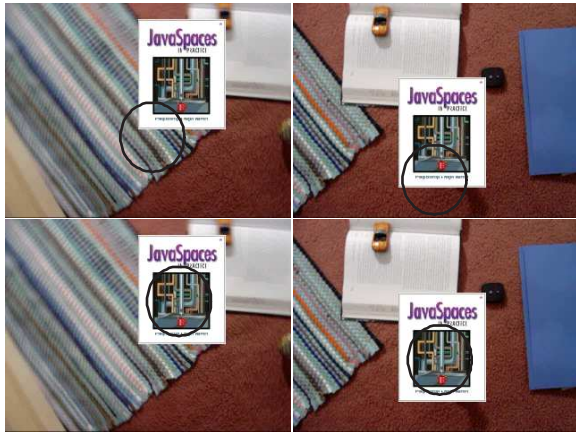


Fig. 2. The tracking results using MKT-SSD [3] (1st row) and our approach (2nd row) respectively.



Fig. 3. Tracking results using KF [1], [5] (black), MKT-SSD [3] (green) and our approach (white) for multiple object tracking in a crowded scene. The first image is the initial frame.

and could not observe the change of object's scale. On the contrary, MKT-SSD could handle object's scale change and the tracking results are more accurate when there is no occlusion and fast motion. However, both of them suffered from the "singularity" problem and failed to track object robustly and consistently. Our approach could 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," *IEEE Trans. Pattern Anal. Machine In-*

tell., 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] Y. Bar-Shalom and T. Fortmann, *Tracking and Data Association*, Academic Press, 1988.
- [6] A. G. Ramm, *Inverse Problem*, Springer, 2005.
- [7] Ken Dutton, Steve Thompson, and Bill Barraclough, *The Art of Control Engineering*, Prentice Hall, 1997.