Correspondence

Robust Control-Based Object Tracking

Wei Qu, *Member, IEEE*, and Dan Schonfeld, *Senior Member, IEEE*

***Abstract*—This correspondence presents a video tracking framework using control-based observer design. It unifies several kernel-based approaches into a consistent theoretical framework by modeling tracking as a recursive inverse problem. The framework relies on observability theory to handle the "singularity" problem and provides explicit criteria for kernel design and dynamics evaluation.**

***Index Terms*—Object tracking, video analysis, video tracking.**

I. Introduction and Related Work

Object tracking from videos has received a significant amount of attention in recent years motivated by its wide application in fields such as video surveillance, human activity analysis, and computer animation. Kernel-based tracking [2] has been demonstrated to provide improved performance compared to other tracking approaches such as optical flow and particle filtering due to its lower computational cost. However, in video sequences containing "complex" scenes such as scale changes, fast motion, or occlusion, the basic kernel-based tracking technique suffers from the well-known *"singularity"* problem [3], [4] 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 for its best matches by optimization schemes such as mean shift [2]. Earlier efforts to handle the "singularity" problem have focused on both aspects. Collins [3] proposed to use a scale kernel to recover object scale changes. Multiple spatially distributed kernels were used to increase the tracker's sensitivity by Hager *et al.* [4]. This approach was further developed by Fan *et al.* [5] who used multiple kernels to enhance the "kernel observability" for articulated objects. Despite the recent progress in the use of multiple kernels for object tracking, the underlying principle of kernel design, required for the solution of the "singularity" problem, remains unknown.

Implicit in all approaches to object tracking is the solution of an inverse problem: determine the state of the tracked object from the observations. The theory of inverse problems [6] has been applied in many applications. Earlier efforts in object tracking relied on elements of control theory to provide a solution to this inverse problem. In particular, a state space model representing the state process and measurement process was hypothesized based on physical and statistical models. For example, a kernel-based target localization integrated with a Kalman filter has been presented in [2]. The Kalman filter used is relatively simple and just an illustration to other procedures integrated with mean shift, where the system and measurement matrices are assumed to be known. Unlike the classical formulation of the observation process, a linear observation process is derived from the kernel equation in [2], [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 [4]. This approach is extended by relying on regularization theory to provide a solution to the constrained linear equation for articulated objects in [5].

Articulated object tracking is a challenging task because of the exponentially increased computational complexity in terms of the degrees of freedom of the object and the severe image ambiguities incurred by the similar neighbor and frequent self-occlusions. Many approaches have been studied to circumvent the problems inherent in articulated object tracking [7], [8]. Various methods have been developed for articulated object tracking using particle filtering [9], [10]. Although these methods are very effective in solving various aspects of articulated object tracking, they still suffer from the high computational complexity. Kernel-based methods have a very promising potential to be exploited for this challenging problem because of their low computational cost. However, the approach needed to apply kernel-based methods for articulated object tracking, especially in the context of an effective solution to the "singularity" problem, remains unresolved.

In this correspondence, we extend the approach presented in [4] and propose a new approach to robust control-based object tracking. The main contributions of our approach are as follows. 1) By formulating object tracking as a recursive inverse problem, the proposed approach provides a unified mathematical framework for a class of methods used for object tracking. 2) We view the linear equation derived from several approaches to kernel-based object tracking as a measurement process and further introduce state dynamics to augment the linear equation for video tracking applications. 3) The proposed framework relies on observability theory from control system to handle the "singularity" problem and thus provides an explicit criterion for both kernel-design and dynamics evaluation in visual object tracking.

The correspondence is organized as follows. We first present the formulation of the object tracking problem as an inverse problem in Section II. Section III discusses several schemes to improve the tracking observability. In Section IV, we propose a control-based observer design for object tracking. In Section V, we extend this approach by presenting a paradigm of control-based observer design for articulated object tracking. Experimental results are provided in Section VI.

II. Object Tracking as an Inverse Problem

Object tracking in video sequences can be defined as an inverse problem: let $\mathbf{x}_t$ represent the object's state (e.g., 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

$$\mathbf{z}_t = g_t(\mathbf{x}_t), \quad g_t : \mathbb{X} \to \mathbb{Z} \tag{1}$$

where $\mathbb{X}$ and $\mathbb{Z}$ are Banach spaces and g_t is a linear/nonlinear operator. The inverse problem is to determine the state $\mathbf{x}_t$ from observation $\mathbf{z}_t$,

Manuscript received November 10, 2007; revised May 2, 2008. Published August 13, 2008 (projected). Portions of this paper have been presented at the IEEE International Conference on Image Processing 2006. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Kai-Kuang Ma.

W. Qu is with Siemens Medical Solutions USA, Inc., Hoffman Estates, IL 60196 USA (e-mail: quweiusa@gmail.com; weiqu@siemens.com).

D. Schonfeld is with the Multimedia Communications Laboratory, Department of Electrical and Computer Engineering, University of Illinois at Chicago, Chicago, IL 60607 USA (e-mail: dans@uic.edu).

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

Digital Object Identifier 10.1109/TIP.2008.2001391

1057-7149/$25.00 © 2008 IEEE

namely, the inverse of operator g_t^{-1}. The *"singularity"* problem discussed in the introduction is also called the *"ill-posed"* problem in the theory of inverse problems [6] where if g_t^{-1} does not exist, the solution of (1) can not be uniquely determined.

If g_t is nonlinear, it is usually hard to get an analytic solution of (1). We can approximate g_t by a linear operator. The linearization can be achieved by dropping higher order terms of the Taylor series

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

Thus, we have the linear observation equation

$$\tilde{\mathbf{z}}_t = \mathbf{C}_t \tilde{\mathbf{x}}_t \tag{3}$$

where $\tilde{\mathbf{z}}_t = \mathbf{z}_t - g_t(\mathbf{x}_0)$, $\tilde{\mathbf{x}}_t = \mathbf{x}_t - \mathbf{x}_0$, and $\mathbf{C}_t = (1/2)g_t'(\mathbf{x}_0)$.

Since the observation $\mathbf{z}_t$ is noisy due to measurement error, the solution of (1) can be estimated by using the least-square-error optimization method

$$\hat{\mathbf{x}}_t = \arg\min_{\mathbf{x}_t} \|\mathbf{z}_t - g_t(\mathbf{x}_t)\|^2 . \tag{4}$$

A method used to derive a linear observation equation for kernel-based tracking has been introduced in [4]. We extend this approach to represent any object tracking problem as follows. Consider a cost function for object tracking

$$\rho\left[\mathbf{q}_t(\mathbf{x}_0), \mathbf{p}_t(\mathbf{x}_t)\right] = \left\| \sqrt{\mathbf{q}_t(\mathbf{x}_0)} - \sqrt{\mathbf{p}_t(\mathbf{x}_t)} \right\|^2 \tag{5}$$

where $\|\cdot\|^2$ is the Matusita metric, $\mathbf{q}_t(\mathbf{x}_0)$ is object's prior model, $\mathbf{p}_t(\mathbf{x}_t)$ is a function of candidate object region. For example, $\mathbf{q}(\cdot)$ and $\mathbf{p}(\cdot)$ can be feature histogram, template representation, or probability density etc. Let $\mathbf{z}_t = \sqrt{\mathbf{q}_t(\mathbf{x}_0)}$, $g_t = \sqrt{\mathbf{p}_t(\mathbf{x}_t)}$, it can be proved that the optimal solution of cost function (5) is the same as the solution of the linear equation $\tilde{\mathbf{z}}_t = \mathbf{C}_t\tilde{\mathbf{x}}_t$, where $\tilde{\mathbf{z}}_t = \sqrt{\mathbf{q}_t(\mathbf{x}_0)} - \sqrt{\mathbf{p}_t(\mathbf{x}_0)}$, the new state $\tilde{\mathbf{x}}_t = \mathbf{x}_t - \mathbf{x}_0$, and $\mathbf{C}_t = (1/2)(\mathbf{p}_t(\mathbf{x}_0))^{-(1/2)}\mathbf{p}_t'(\mathbf{x}_0)$.

When limiting the state to be the object's center, i.e. $\mathbf{x} = c$, and using a kernel-based color histogram for $\mathbf{q}(\cdot)$ and $\mathbf{p}(\cdot)$, it can be shown that $\mathbf{q}(c_0) = \mathbf{U}^T\mathbf{K}(c_0)$, $\mathbf{p}(c) = \mathbf{U}^T\mathbf{K}(c)$, and $\mathbf{C} = (1/2)\mathrm{diag}[\mathbf{p}(c_0)]^{-(1/2)}\mathbf{U}^T\mathbf{J}_K(c_0)(c - c_0)$, where c_0 is object's prior center, $\mathbf{U}$ is a sifting matrix indicating which object pixel belong to which bins, $\mathbf{K}$ is a vector of the kernel function, $\mathbf{J}_K$ is the Jacobian matrix of kernel vector $\mathbf{K}$, and $\mathrm{diag}[\mathbf{p}]$ represents the matrix with $\mathbf{p}$ on its diagonal [4]. This problem has been solved by using a Newton-style approach with sum of squared differences (SSD) in [4].

III. Improving Tracking Observability

Solving the inverse problem of (3) is not trivial. The dimensionality of the 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 $\mathbf{C}_t$. The primary difficulty with *"ill-posed"* problems is that the state is undetermined due to small (or zero) singular values of $\mathbf{C}_t$. In other words, if the $\mathrm{rank}(\mathbf{C}_t) < n$, where n is the dimensionality of state $\tilde{\mathbf{x}}$, then $\mathbf{C}_t^{-1}$ does not exist. Although originally the earlier efforts of kernel-based approaches presented in [3]–[5] did not aim to solve the "ill-posed" problem, these efforts can be interpreted as enhancing the $\mathrm{rank}(\mathbf{C}_t)$ by the following ways.

1) **Using multiple kernels to enhance the $\mathrm{rank}(\mathbf{C}_t)$**
It has been shown in [3] and [4] that multiple kernels can improve the tracking performance. However, it is not clear why multiple kernels work better than only one in theory and how multiple kernels should be designed. By formulating object tracking as an inverse problem, we can now answer these questions explicitly: $\tilde{M}$ kernels can construct $\tilde{M}$ observation equations, $\mathbf{z}_t^{\tilde{m}} = \mathbf{C}_t^{\tilde{m}}x_t$, $\tilde{m} = 1,\ldots,\tilde{M}$. By combining them in different ways, the $\mathrm{rank}(\mathbf{C}_t)$ has the potential to increase. For example, in [4], a "stacked system" constructed from multiple kernels is exploited. In this case, the system observation matrix is constructed by concatenating the sub-matrices together in a column $\mathbf{C}_t = [\mathbf{C}_t^1,\ldots,\mathbf{C}_t^{\tilde{M}}]^T$, we observe that $\mathrm{rank}(\mathbf{C}_t) \geq \mathrm{rank}(\mathbf{C}_t^{\tilde{m}})$. Thus, the principle of kernel design is that additional kernels should help to increase the $\mathrm{rank}(\mathbf{C}_t)$.

2) **Tikhonov regularization to enhance the $\mathrm{rank}(\mathbf{C}_t)$.**
A kernel-based method using joint state representation and a length constraint among states has been presented in [5] for articulated object tracking. Different constraints such as length constraint or skeleton model [11] have been used for articulated object tracking. We propose to improve the tracking observability for any constraints by using the well-known *Tikhonov regularization* [6]. To cope with the *"ill-posed"* problem, 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

$$\hat{\mathbf{x}} = \arg\min\left\{\|\mathbf{z}_t - \mathbf{C}\mathbf{x}_t\|^2 + \lambda\|\mathbf{b} - \mathbf{G}\mathbf{x}_t\|^2\right\} \tag{6}$$

where the regularization parameter $\lambda > 0$, $\mathbf{G}\mathbf{x}_t = \mathbf{b}$ represents additional constraints.

By using generalized singular value decomposition, it can be shown that (6) has a solution provided by a linear equation [6]

$$\mathbf{C}_t^T\mathbf{z}_t + \lambda\mathbf{G}^T\mathbf{b} = \left(\mathbf{C}_t^T\mathbf{C}_t + \lambda\mathbf{G}^T\mathbf{G}\right)\mathbf{x}_t. \tag{7}$$

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

Although multiple kernels and regularization have shown the potential to improve the tracking performance, in our experiments, we have found that in the case of fast motion or occlusions, the kernel-based approaches still suffer from the *"ill-posed"* problem and can not track the object robustly. These limitations have motivated us to formulate object tracking as a recursive inverse problem which includes the object's state dynamics and relies on an optimal observer from control theory to solve the *"ill-posed"* problem.

IV. 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

$$\mathbf{x}_{t+1} = \mathbf{A}_t\mathbf{x}_t + \mathbf{w}_t \tag{8}$$

$$\mathbf{z}_t = \mathbf{C}_t\mathbf{x}_t + \mathbf{v}_t \tag{9}$$

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 a Kalman filter [12]. A method combining Kalman filter and a kernel-based target model was presented in [2], where 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 are usually unknown and may be time-variant. Different motion estimation techniques and scene-based prior knowledge can be used to approximate the state dynamics. To select the best motion estimate for handling the *"ill-posed"* problem, we need a criterion to evaluate which one can yield the largest possibility of uniquely "observing" the state? This observation inspired us to introduce *observability theory* from control systems [12]. It has been proved that the observability of a linear system described by (8) and (9) can be determined as follows [12].

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

This theorem is consistent with our earlier analysis for nonrecursive inverse problems, where the observability matrix $\mathcal{O}_t$ degrades to matrix $\mathbf{C}_t$ since there is no state equation. In this case, we can prove that the rank of the observability matrix with augmented state dynamics $\mathbf{A}_t$ increases compared to the rank of the observability matrix of the original inverse problem, i.e. $\mathrm{rank}(\mathcal{O}_t) \geq \mathrm{rank}(\mathbf{C}_t)$. This result can be easily shown since the observability matrix of the original inverse problem $\mathbf{C}_t$ is a submatrix of the observability matrix with augmented state dynamics $\mathbf{A}_t$.

The higher the rank of the observability matrix, the higher observability of the system. Thus, the recursive system can cope with the *"ill-posed"* problem better than traditional solutions to inverse problems that do not rely on state dynamics. Guided by the observability theorem, we provide two paradigms for the solution of the recursive inverse problem using control-based observers in the following section. The effectiveness of the proposed approach is demonstrated in our experiments as shown in Section VI.

V. Two Paradigms of Control-Based Observer Design

A. A Paradigm for Single Object Tracking

In this section, we give a paradigm of the proposed control-based observer design for single object tracking. We first introduce a brief review of the basic concept of kernel-based object tracking [2]–[4]. These steps construct the observation equation (9). Then, we design the state equation (8) based on the **Observability Theorem**.

Given a set of n_p pixels $\{\tilde{s}_1, \ldots, \tilde{s}_{n_p}\}$ in the image region, the kernel-based histogram $\mathbf{q} = [q_1, q_2, \ldots, q_{m_b}]^T \in \Re^{m_b}$ can be constructed as follows: First, denote $u = 1, \ldots, m_b$ as the feature bins. Then, let each pixel $\tilde{s}_i$ fall into one of the m_b bins according to some predefined feature, say, color. Notice that we denote i as the index of pixels here. This can be expressed by a mapping function $b(\tilde{s}_i)$. Finally, we use a kernel $K(\tilde{s}_i)$ to spatially weight the pixels. Therefore, the kernel-based histogram is given by

$$q_u = \frac{1}{\eta}\sum_{i=1}^{n_p} K(\tilde{s}_i - c_0)\delta\left(b(\tilde{s}_i), u\right) \tag{10}$$

where δ is the Kronecker delta function, c_0 is a prior kernel center, and $\eta = \sum_{j=1}^{n_p} K(\tilde{s}_j - c_0)$ is a normalization factor which ensures that $\sum_{u=1}^{m_b} q_u = 1$.

The above equation can be compactly written in matrix form as

$$\mathbf{q}(c_0) = \mathbf{U}^T\mathbf{K}(c_0) \tag{11}$$

where

$$\mathbf{U} = \begin{bmatrix} \delta\left(b(\tilde{s}_1, u_1)\right) & \ldots & \delta\left(b\left(\tilde{s}_1, u_{m_b}\right)\right) \\ \vdots & \ddots & \vdots \\ \delta\left(b\left(\tilde{s}_{n_p}, u_1\right)\right) & \ldots & \delta\left(b\left(\tilde{s}_{n_p}, u_{m_b}\right)\right) \end{bmatrix}$$

and

$$\mathbf{K} = \begin{bmatrix} K(\tilde{s}_1 - c_0) \\ \vdots \\ K\left(\tilde{s}_{n_p} - c_0\right) \end{bmatrix}$$

and $\mathbf{U} \in \mathbb{R}^{n_p\times m_b}$, $\mathbf{K} \in \mathbb{R}^{n_p\times 1}$.

Similarly, we can calculate the kernel-based feature histogram for the candidate region centered at c as

$$\mathbf{p}(c) = \mathbf{U}^T\mathbf{K}(c). \tag{12}$$

Consider a cost function

$$O(c) = \|\sqrt{\mathbf{q}_t(c_0)} - \sqrt{\mathbf{p}_t(c_t)}\|^2. \tag{13}$$

After linearization w.r.t. Δc, we can get

$$\mathbf{z}_t = \mathbf{C}_t\mathbf{x}_t \tag{14}$$

where

$$\begin{aligned} \mathbf{z}_t &= \sqrt{\mathbf{q}_t(c_0)} - \sqrt{\mathbf{p}_t(c_0)} \\ \mathbf{x}_t &= c_t - c_0 \text{ and} \\ \mathbf{C}_t &= \frac{1}{2}\left(\mathbf{p}_t(c_0)\right)^{-\frac{1}{2}}\mathbf{p}_t'(c_0). \end{aligned} \tag{15}$$

Moreover, guided by the **Observability Theorem**, we can estimate the dynamic matrix for the state equation. Prior knowledge and motion estimation techniques can be used to estimate a set of dynamics $\{\mathbf{A}_t^1, \ldots, \mathbf{A}_t^J\}$. Unless they are predefined by prior knowledge, these sub-matrices are unknown in advance and may change at distinct time for the situation when some new motion estimation techniques can be exploited or some estimates are not valid anymore. Then, at each time, we dynamically select the optimal dynamic matrix $\mathbf{A}_t^j$ which is designed to reach the maximal observability, i.e., maximum $\mathrm{rank}(\mathcal{O}_t)$. This process ensures that the observer has a higher possibility to determine the state uniquely. The optimal dynamic matrix $\mathbf{A}_t^j$ can be used to construct a *Kalman–Bucy* filter and the estimate of the state is given by [12]

$$\hat{\mathbf{x}_t} = [\mathbf{I} - \mathbf{L}_t\mathbf{C}_t]\mathbf{A}_t\mathbf{x}_{t-1} + \mathbf{L}_t\mathbf{z}_t \tag{16}$$

where $\mathbf{I}$ is the identity matrix; $\mathbf{A}_t$ is the state dynamics, which can be dynamically estimated; $\mathbf{C}_t$ is the observation matrix in (14); $\mathbf{L}_t$ is the filter gain matrix given by

$$\mathbf{L}_t = \mathbf{P}_{t-1}\mathbf{C}_t^T[\mathbf{C}_t\mathbf{P}_{t-1}\mathbf{C}_t^T + \mathbf{R}_t]^{-1} \tag{17}$$

where $\mathbf{P}_{t-1}$ is the estimation covariance at time $t-1$, $\mathbf{R}_t$ is the covariance of the observation noise $\mathbf{v}_t$. For implementation details of the Kalman–Bucy filter, we refer the reader to [12]. We will refer to the proposed approach as control-based object tracking (CBOT).

When none of the estimated dynamic matrices in the set $\{\mathbf{A}_t^1, \ldots, \mathbf{A}_t^J\}$ is full rank and several of them, say, two, $\mathbf{A}_i$ and $\mathbf{A}_j$ have the highest rank, our approach would select the current tracking result by averaging the results of both motion models. The two estimated motion matrices will be kept in the following frames until one of them is no longer the highest rank matrix. This scheme allows us to keep multiple hypotheses when there is no dynamic model that results in complete disambiguation of the tracked object's state.

B. A Paradigm for Articulated Object Tracking

In this section, we present a paradigm of the proposed approach for articulated object tracking. Suppose we have M joints needed to track an articulated object. We denote their region centers as $\{c_1, \ldots, c_M\}$. By assuming that these joints are restricted by a certain structure constraint $\rho(c_1, \ldots, c_M) = 0$, for example, the skeleton model [11], the objective function of the tracking problem is given by

$$\begin{aligned} O(c_1, c_2, \ldots, c_M) = \sum_{m=1}^{M}\left\|\sqrt{\mathbf{q}_m\left(c_m^0\right)} - \sqrt{\mathbf{p}_m(c_m)}\right\|^2 \\ +\lambda\left\|\rho(c_1, \ldots, c_M)\right\|^2 \end{aligned} \tag{18}$$

where c_m^0 is a prior kernel center for joint m.

After linearization w.r.t. $\{\Delta c_1, \Delta c_2, \ldots, \Delta c_M\}$, we have

$$\begin{aligned} \mathbf{z}_t &= \mathbf{C}_t \mathbf{x}_t \\ \mathbf{b} &= \mathbf{G}_t \mathbf{x}_t \end{aligned} \tag{19}$$

where

$$\begin{aligned} \mathbf{x}_t &= [\Delta c_1, \ldots, \Delta c_M]^T, \quad \mathbf{C}_t = \begin{bmatrix} \mathbf{C}_1 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & \mathbf{C}_M \end{bmatrix} \\ \mathbf{z}_t &= \left[\sqrt{\mathbf{q}_1} - \sqrt{\mathbf{p}(c_1)}, \ldots, \sqrt{\mathbf{q}_M} - \sqrt{\mathbf{p}(c_M)}\right]^T \\ \mathbf{b} &= -\rho(c_1, c_2, \ldots, c_M), \text{ and} \\ \mathbf{G} &= \left[\frac{\partial \rho}{\partial c_1}, \ldots, \frac{\partial \rho}{\partial c_M}\right]. \end{aligned} \tag{20}$$

By further stacking the two observation equations in (19) together, we have a new system observation equation

$$\tilde{\mathbf{z}} = \tilde{\mathbf{C}}\tilde{\mathbf{x}} \tag{21}$$

where $\tilde{\mathbf{z}} = [\mathbf{z}, \mathbf{b}]^T$ and $\tilde{\mathbf{C}} = [\mathbf{C}, \mathbf{G}]^T$.

We note that unlike traditional Kalman filter tracking approaches [2], the state equation in our model is not predetermined. Instead, the dynamics of the state equation are selected to maximize observability and thus solve the "singulaity" problem. Similar to our approach to the previous paradigm for single object tracking, we can evaluate the estimated system dynamics by using the **Observability Theorem**. Specifically, at each time t, we select the best state dynamic matrix $\mathbf{A}_t^{j,m}$, for each articulated object and thus construct the system dynamic matrix $\mathbf{A}_t = [\mathbf{A}_t^1, \ldots, \mathbf{A}_t^M]^T$. We subsequently rely on the *Kalman–Bucy* filter to estimate the state.

VI. Experimental Results

The performance of the proposed CBOT has been thoroughly demonstrated on a set of different image sequences including both both synthetic and real-world video. The videos were captured by a resolution of 320×240 pixels with a frame rate of 30 fps. In all of the experiments, we used a multiple kernel-based color histogram similar to the MKT-SSD approach [4], which has ten bins for each RGB channel, respectively. More bins for each channel may yield more reliable tracking results while sacrificing computational speed. We limit ourselves to a 30-bin histogram in the experiments is only to conduct a fair comparison of the various tracking techniques.

A. Qualitative Tracking Results

The synthetic video has a book moving according to predefined state dynamics in a cluttered scene. The changing background prevents the use of background subtraction. We use this video to demonstrate the performance of the proposed approach for single object tracking with fast motion and severe background clutter. We have tested our approach and MKT-SSD [4] on sequences with original frame rate of 30fps and a lower frame rate of 10fps, where the object's motion becomes much faster. Only the observation equation, but not the constraint equations in (19), is used for single object tracking. Fig. 1 presents the tracking trajectories of the object's center. 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 fast motion presented in the 10-fps sequence. However, due to exploiting the state dynamics and coping with the "ill-posed" problem, our approach achieved a more robust performance in both cases. Fig. 2 illustrates the tracking results for the 10-fps sequence.

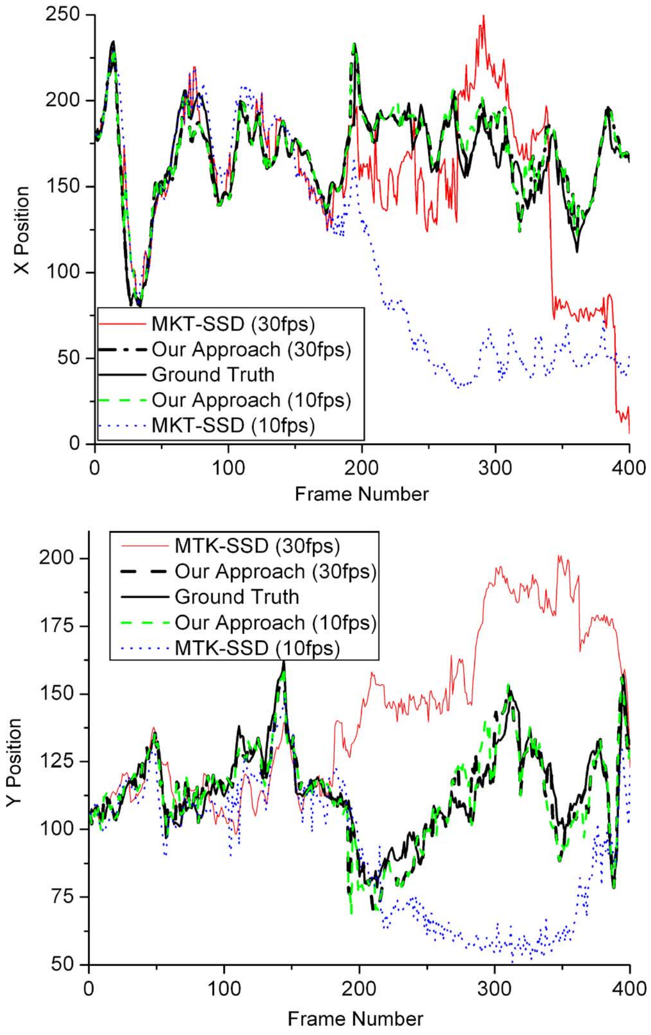
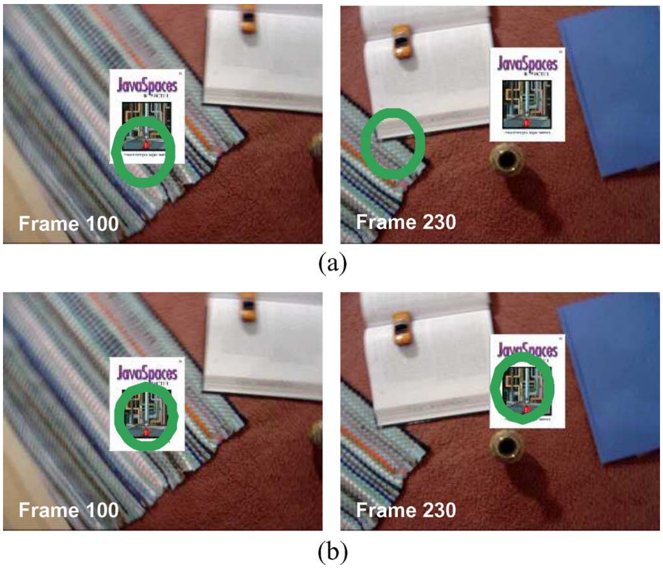


Fig. 1. Tracking trajectories of object's center using our CBOT and MKT-SSD [4] for the synthetic sequence with different frame rates.



(a)

(b)

Fig. 2. Tracking results using MKT-SSD [4] and our CBOT, respectively. (a) MKT-SSD; (b) CBOT.

We further compare the performance of our CBOT with MKF-SSD [4] and the regular Kalman filter (KF) tracking approach [2] on a real-

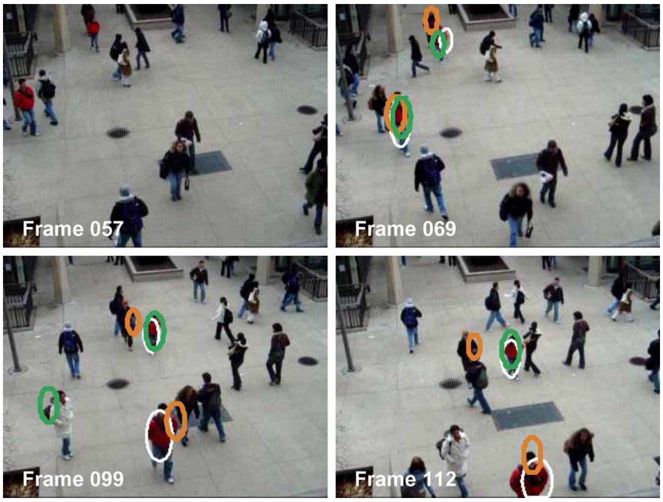
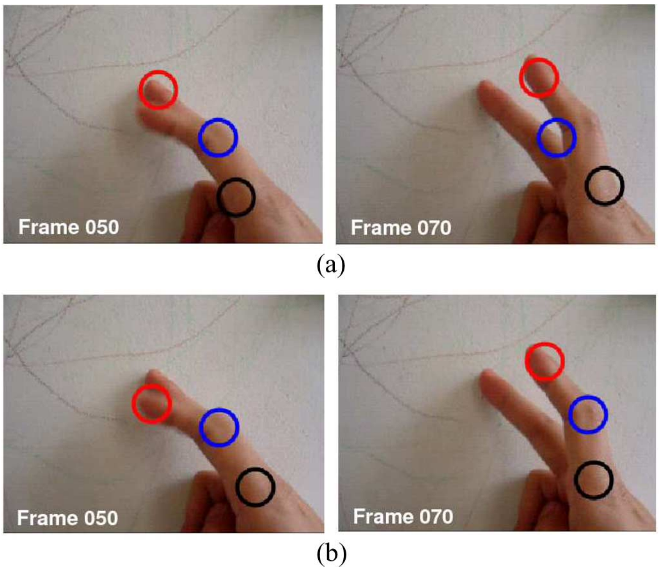


Fig. 3. Tracking results using KF [2] (red), MKT-SSD [4] (green) and our CBOT (white) for the sequence `Plaza`. The first image is the initial frame.

world video sequence, `Plaza`. This video has a crowded scene presenting various motions and different occlusions. Because of the clutters in the scene and different measurement errors, all tracking methods performed in a noisy environment. We use two independent trackers for two pedestrians with remarkable color features. Two matrices $\mathbf{A}_1$ and $\mathbf{A}_2$ were used for 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 we used ellipses of different colors to show the results of the different approaches. As we can see, KF (red) is helpful to handle fast motion. However, this approach suffered from the background clutter and could not observe the change of the object's scale. MKT-SSD could handle the object's scale change and the tracking results were more accurate when there was no occlusion and fast motion. However, both of these approaches failed to track objects robustly and consistently. Our approach could achieve more robust tracking results by handling both partial occlusion and fast motion in the crowded scene.

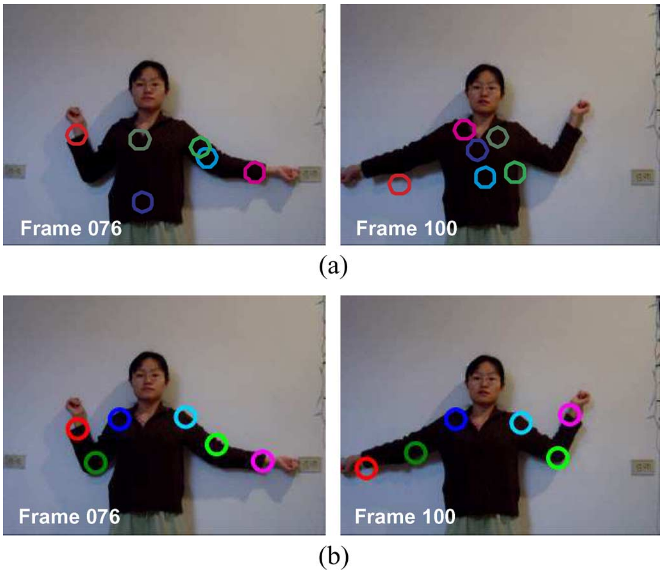
The performance of the proposed CBOT was also compared with MKT-SSD on different videos for articulated object tracking. The `2FINGERS` sequence contains two fingers, one static in the back, and one moving rapidly in the front. We tracked three joints of the front finger and regarded the back one as background clutter. Length constraints [11] are used between adjacent joints. Two matrices $\mathbf{A}_1$ and $\mathbf{A}_2$ were used for the state dynamics, where $\mathbf{A}_1$ was assumed to be an identity matrix and $\mathbf{A}_2$ was estimated by motion extraction from optical flow method. Fig. 4 shows the tracking results of MKT-SSD and CBOT. It can be seen that MKT-SSD could not produce satisfactory results. When the finger moved fast, some trackers falsely attached to the back finger. By exploiting the state dynamics, CBOT could handle fast motion and achieved very robust results against the background clutter. We also tested MKT-SSD and CBOT on the `GIRL` video. A length constraint model was first learned from the training sample images and then exploited in CBOT. Fig. 5(a) shows sample frames using MKT-SSD. As we can see, the results were not stable, especially when the arms moved close to the torso. The image ambiguities make the trackers failed. By embedding length constraints between adjacent joints and introducing an adaptive state equation based on the observability theorem, our CBOT could provide a more stable tracking of the joints, even in strong neighborhood clutters, partial occlusion and fast movement as shown in Fig. 5(b).



(a)

(b)

Fig. 4. Comparison of MKT-SSD [4] and the proposed CBOT for the sequence `2FINGERS`.



(a)

(b)

Fig. 5. Comparison of MKT-SSD [4] and the proposed CBOT for the sequence `GIRL`.

B. *Quantitative Performance Analysis and Comparisons*

How to quantitatively evaluate the performance of articulated object tracking remains an open problem. We compare the tracking accuracy of different approaches by defining the *false position rate* (FPR) and *false label rate* (FLR) given by

$$\mathrm{FPR} = \frac{\text{The number of position failures}}{\text{The total number of articulated object parts}}$$
$$\mathrm{FLR} = \frac{\text{The number of label failures}}{\text{The total number of articulated object parts}}$$

where a *position failure* is defined as the absence of a tracker associated with one of the tracked parts; and a *label failure* is defined as a tracker associated with a false object part.

TABLE I
SPEED AND ACCURACY COMPARISONS ON THE GIRL SEQUENCE

Method	Speed (fps)	FPR	FLR
MKT-SSD	$17 \sim 17.5$	25.3%	23.4%
CBOT	$10 \sim 11.6$	2.2%	2%

In Table I, we present the speed and accuracy data of MKT-SSD and the proposed CBOT on the GIRL sequence, which has 466 frames. Compared with MKT-SSD, CBOT requires a higher computational cost to calculate the state dynamics and thus decreases the running speed. However, the performance of the proposed CBOT was much more robust than MKT-SSD. By exploiting the motion information, CBOT could handle fast motion, strong image ambiguities and neighborhood clutters, and partial occlusions where MKT-SSD failed. Thus, both the FPR and FLR of CBOT decreased for both single and articulated object tracking.

VII. CONCLUSION

In this correspondence, we presented a control-based object tracking approach. It unifies several existing object tracking methods into a consistent theoretical framework by modeling object tracking as a recursive inverse problem. It handles the "singularity" problem by using a control-based optimal observer design and provides an explicit principle for kernel design and dynamics evaluation. Experimental results have shown the superior performance of the proposed approach compared with existing kernel-based tracking methods.

Several important problems remain unresolved and will be the subject of future research. 1) Due to only using the simple length constraint between the adjacent joints, the current implementation still has limitations in solving severe self occlusions for articulated object tracking. We plan to include more sophisticated constraints to improve the tracking performance. 2) It is critical to allow the design of the observation equations to include additional prior interaction information. We plan to investigate this issue and extend the proposed framework to multiple object tracking with severe occlusions. 3) The current formulation of the proposed CBOT method is based on a joint state representation. Analysis of CBOT in the context of a distributed framework could be used to improve the computational efficiency of various kernel-based tracking algorithms.

ACKNOWLEDGMENT

The authors would like to thank M. Dewan for sharing part of the code used for implementing MTK-SSD [4].

REFERENCES

[1] W. Qu and D. Schonfeld, "Robust kernel-based tracking using optimal control," presented at the IEEE Int. Conf. Image Processing, Atlanta, GA, 2006, Best Student Paper Award.

[2] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 564–577, May 2003.

[3] R. T. Collins, "Mean-shift blob tracking through scale space," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2003, vol. 2, pp. 234–240.

[4] G. D. Hager, M. Dewan, and C. V. Stewart, "Multiple kernel tracking with *SSD*," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2004, vol. 1, pp. 790–797.

[5] Z. Fan, Y. Wu, and M. Yang, "Multiple collaborative kernel tracking," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2005, vol. 2, pp. 502–509.

[6] A. G. Ramm*, Inverse Problem*. New York: Springer, 2005.

[7] M. H. Lin, "Tracking articulated objects in real-time range image sequences," presented at the Int. Conf. Computer Vision, Corfu, Greece, 1999.

[8] T. Drummond and R. Cipolla, "Real-time tracking of highly articulated structures in the presence of noisy measurements," presented at the Int. Conf. Computer Vision, Vancouver, BC, Canada, 2001.

[9] L. Sigal, S. Bhatia, S. Roth, M. J. Black, and M. Isard, "Tracking loose-limbed people," presented at the IEEE Conf. Computer Vision and Pattern Recognition, Washington, DC, 2004.

[10] W. Qu and D. Schonfeld, "Real-time decentralized articulated motion analysis and object tracking from videos," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2129–2138, Aug. 2007.

[11] Z. Chen and H. Lee, "Knowledge-guided visual perception of 3D human gait from a single image sequence," *IEEE Trans. Syst. Man. Cybern.*, vol. 22, no. 2, pp. 336–342, Feb. 1992.

[12] K. Dutton, S. Thompson, and B. Barraclough*, The Art of Control Engineering*. Englewood Cliffs, NJ: Prentice-Hall, 1997.