Multiple Collaborative Kernel Tracking

Zhimin Fan Ying Wu Ming Yang
Department of Electrical & Computer Engineering
Northwestern University, 2145 Sheridan Road, Evanston, IL 60208
{zfa825,yingwu,mya671}@ece.northwestern.edu

Abstract

This paper presents a novel multiple collaborative kernel approach to visual tracking. This approach treats kernel-based tracking in a more general setting, i.e., a relaxation and constraints formulation, in which a complex motion is represented by a set of inter-correlated simpler motions. With this formulation, we present a rigorous analysis on a critical issue of kernel observability and obtain a criterion, based on which we propose a new method using collaborative kernels that has the theoretical guarantee of enhanced observability. This new method has been shown to be computationally efficient in both theory and practice, which can be readily applied to complex motions such as articulated motions.

1. Introduction

Kernel-based methods [2, 9] have attracted much attention in computer vision [4, 6, 10] and have recently shown promising performance in the challenging problem of visual tracking [5]. In this context, the representation of the object being tracked is the convolution of the object features with a spatially weighted kernel, which enables efficient gradient based optimization methods, such as mean shift [4] or Newton-style method [7], to search for the best match to the target model based on the collected visual measurements (or observations). Thus, one of the most appealing merits of kernel-based trackers is their low computational cost, compared with other commonly employed tracking schemes, such as particle filters [8] or exhaustive template matching.

Since the kernel-based tracking methods are gradient-based differential approaches, their performances are largely effected by the quality of the searching directions calculated from the measurements (i.e., the discrepancy between the candidates and the target model). In practice, we observe a singular case where the searching direction is indifferent to the measurements, i.e, the measurements become more or less invariant to some motion parameters, such that these motion parameters are not recoverable or observable. Therefore, three critical issues of both theoretical and practical importance need to be investigated:

- Is there a criterion or a test that detects such singularities and checks the observability of the motion?
- Is there a principled way of kernel design to prevent or alleviate such singularities?
- Can we cope with such singularities in more complex motions (e.g., articulation) while still achieving computational efficiency?

There have been some initial studies related to these questions. For example, in [3], to deal with the problem that most kernels, being scale-invariant, cannot recover the scale changes of the target, a method was proposed to combine multiple kernels of different resolutions. An outstanding initial investigation on multiple kernels was presented in [7], where an unconstrained linear least square formulation was given and the motion singularity can be revealed by the rank deficiency when approaching its solution, based on which a multiple kernel method was proposed to possibly reduce the risk of rank deficiency.

These initial investigations on multiple kernels are meaningful, but they are inadequate. For example, although several suggestions have been made in [7] on designing multiple kernels, it is desirable to have a more rigorous theoretical guarantee on motion recoverability and a more principled and generalizable approach to kernel design. In addition, complex motions (e.g., motions of articulated bodies) pose a great challenge to most existing kernel-based tracking algorithms which are largely confined by single target and simple motions, and this is a topic remained largely unexplored among the literatures of kernel-based methods. Although many top-down sampling-based algorithms have been explored for complex motion [1, 11], they are in general computationally demanding. Thus, it will be very meaningful if the bottom-up kernel-based solutions can be found.

Inspired by [7, 11], we present a novel multiple collaborative kernel approach to visual tracking in this paper. This approach treats kernel-based tracking in a more general formulation, i.e., a relaxation and constraints formulation, in which a complex motion can be represented by a set of inter-correlated simpler motions. In this new formulation, the state equation describes the constraints among these simpler motions, and the measurement equation characterizes the independent visual measurement processes of these simpler motions. With this formulation, this paper presents a rigorous theoretical analysis on the singularity issue, i.e., kernel observability, and presents the observability criterion. Based on this, we propose the multiple collaborative kernel method that has the theoretical guarantee of enhanced observability. This new method has been shown to be computationally efficient in both theory and practice.

Advancing the state of the art, the contributions of this work include: (1) the theoretical results that unify the study of the motion observability issue in most kernel-based methods including single and multiple kernels; (2) a principled way of designing observable kernels, i.e.. the multiple collaborative kernels, that can be easily generalized to complex motions; (3) an efficient computational paradigm to cope with complex motion due to the "collaboration" among a set of inter-correlated kernels, each of which only takes charge of recovering a simpler motion.

The proposed design of "multiple collaborative kernels" closely follows the theoretical concerns on "kernel-observability", i.e., singularity in motion detection, and substantially broadens the applicability of kernel based methods for tracking of multiple targets with complex motions, such as the articulated body motions.

2. Kernel-based Tracking

We first review the basic idea of kernel-based tracking with notations similar to [5, 7].

Assume $\{\mathbf{x}_i\}_{i=1\ldots n}$ be the pixel locations of the target (this can be generalized to more complex motion parameters). For each pixel $\mathbf{x}_i$, a binning function $b(\mathbf{x}_i)$ maps a predefined feature, e.g., the color, of $\mathbf{x}_i$ onto a histogram bin u, with $u \in \{1\ldots m\}$. Let K be a spatially weighting kernel. Then a histogram representation of the target $\mathbf{q}= [q_1, q_2, \ldots, q_m]^T \in \mathbb{R}^m$ can be computed as,

$$q_u = \frac{1}{C}\sum_{i=1}^{n} K(\mathbf{x}_i - \mathbf{c})\delta(b(\mathbf{x}_i), u), \tag{1}$$

where δ is the Kronecker delta function, $\mathbf{c}$ is the kernel center and C is the normalization factor. Its more concise matrix form can be written as [7]:

$$\mathbf{q}(\mathbf{c}) = \mathbf{U}^T\mathbf{K}(\mathbf{c}), \tag{2}$$

where

$$\mathbf{U} = \begin{bmatrix} \delta(b(\mathbf{x}_1, u_1)) & \ldots & \delta(b(\mathbf{x}_1, u_m)) \\ \vdots & \vdots & \vdots \\ \delta(b(\mathbf{x}_n, u_1)) & \ldots & \delta(b(\mathbf{x}_n, u_m)) \end{bmatrix} \in \mathbb{R}^{n\times m},$$

and

$$\mathbf{K} = \frac{1}{C}\begin{bmatrix} K(\mathbf{x}_1 - \mathbf{c}) \\ \vdots \\ K(\mathbf{x}_n - \mathbf{c}) \end{bmatrix} \in \mathbb{R}^n.$$

In general, for the target model, the kernel is centered at $\mathbf{0}$, and we denote it by $\mathbf{q} = \mathbf{U}^T\mathbf{K}$. By the same token, we represent the histogram observed at a given candidate region centered at $\mathbf{c}$ as:

$$\mathbf{p}(\mathbf{c}) = \mathbf{U}^T\mathbf{K}(\mathbf{c}). \tag{3}$$

Given an initial start at location $\mathbf{c}$, the core problem in tracking is to find a best displacement $\Delta\mathbf{c}$ such that the measurements $\mathbf{p}(\mathbf{c} + \Delta\mathbf{c})$ at the new location best matches the target $\mathbf{q}$, i.e.,

$$\Delta\mathbf{c}^* = \arg\min_{\Delta c} O(\mathbf{q}, \mathbf{p}(\mathbf{c} + \Delta\mathbf{c})), \tag{4}$$

where $O(\cdot, \cdot)$ is the objective function for matching. For example, it can be the Bhattacharyya coefficient employed in the mean shift algorithm [5]:

$$O_1(\Delta\mathbf{c}) \overset{\triangle}{=} -\langle\sqrt{\mathbf{q}}, \sqrt{\mathbf{p}(\mathbf{c} + \Delta\mathbf{c})}\rangle = -\sqrt{\mathbf{q}}^T\sqrt{\mathbf{p}(\mathbf{c} + \Delta\mathbf{c})}.$$

In addition, it can also be the Matusita metric used in [7]:

$$O_2(\Delta\mathbf{c}) \overset{\triangle}{=} \|\sqrt{\mathbf{q}} - \sqrt{\mathbf{p}(\mathbf{c} + \Delta\mathbf{c})}\|^2.$$

As indicated in [7], these two choices are equivalent.

Various optimization techniques can be employed to solve the problem in Eq. 4, such as the mean shift procedure [5] or a Newton-style method in [7]. Of course, when the final displacement $\Delta\mathbf{c}^* = \mathbf{0}$, a local optimum is achieved and thus a match to the target is found. In practice, unfortunately, we sometimes are plagued in a singularity situation where the same optimal value of $O(\Delta\mathbf{c})$ can be achieved over a continuous range, i.e., any candidate region induced by the movement in this range matches the target equally well. In other words, the motion parameters can not be uniquely determined, or can not be fully *observed* through the kernel. Inspired by some initial analysis in [7] on this problem of kernel-observability, we present in next section our study on it.

3. Kernel-observability Analysis

The issue of "kernel-observability" mentioned in the previous section can be related to the "system-observability" of a more general system in Eq. 5 for a better definition and explanation. We omit the noise terms for clarity, since it does not affect the analysis.

$$\left\{ \begin{array}{lcl} \Omega(\mathbf{x}) & = & 0 \\ \mathbf{y} & = & \mathcal{M}(\mathbf{x}), \end{array} \right. \tag{5}$$

where $\Omega(\mathbf{x})$ represents the inherent property of the state variable $\mathbf{x}$, such as the complexity, self-contained constraint or the system dynamics, and $\mathcal{M}$ denotes the observation process or measurement process. In this system, the state variable $\mathbf{x}$ is hidden and can only be estimated through the measurement $\mathbf{y}$. In the tracking scenario, the state variable refers to the motion to be estimated. Here we do not limit our discussions only to the 2D displacements, but generalize it to r dimensional motion vector, i.e., $\mathbf{x} \in \mathbb{R}^r$. A critical issue is whether or not $\mathbf{x}$ can be *uniquely determined* from $\mathbf{y}$, i.e., the *observability* of this system.

In the context of kernel tracking, we treat

$$\mathbf{x} \stackrel{\triangle}{=} \Delta\mathbf{c}.$$

Our analysis is based on the linearization of the system at a given initial start $\mathbf{c}$, since the local property of $\mathbf{c}$ is the mostly concerned in the tracking problem. The collected image evidence for $\mathbf{c} + \Delta\mathbf{c}$ is the difference between the target and the candidate, i.e., $\sqrt{\mathbf{q}} - \sqrt{\mathbf{p}(\mathbf{c} + \Delta\mathbf{c})}$. Linearizing it w.r.t. $\Delta\mathbf{c}$, we have

$$\sqrt{\mathbf{q}} - \sqrt{\mathbf{p}(\mathbf{c})} = \mathbf{M}\Delta\mathbf{c},$$

where $\sqrt{\mathbf{q}}$, $\sqrt{\mathbf{p}(\mathbf{c})} \in \mathbb{R}^m$, $\Delta\mathbf{c} \in \mathbb{R}^r$, $\mathbf{M} \in \mathbb{R}^{m \times r}$,

$$\mathbf{M} = \tfrac{1}{2}\texttt{diag}(\mathbf{p}(\mathbf{c}))^{-\frac{1}{2}}\mathbf{U}^T\mathbf{J}_\mathbf{K}(\mathbf{c}),$$

$$\mathbf{J}_\mathbf{K}(\mathbf{c}) = \begin{bmatrix} \nabla_c K(\mathbf{x}_1 - \mathbf{c}) \\ \nabla_c K(\mathbf{x}_2 - \mathbf{c}) \\ \vdots \\ \nabla_c K(\mathbf{x}_n - \mathbf{c}) \end{bmatrix},$$

and $\texttt{diag}(\mathbf{p})$ represents the matrix with $\mathbf{p}$ on its diagonal. This result was actually obtained in [7]. In view of this, we treat the measurement $\mathbf{y} \stackrel{\triangle}{=} \sqrt{\mathbf{q}} - \sqrt{\mathbf{p}(\mathbf{c})}$, and thus the linearized **measurement equation** can be written as:

$$\mathbf{y} = \mathbf{M}\Delta\mathbf{c} = \mathbf{M}\mathbf{x}. \tag{6}$$

When the motion constraints holds at $\mathbf{c} + \Delta\mathbf{c}$, i.e., $\Omega(\mathbf{c} + \Delta\mathbf{c}) = 0$, we can always linearize it as

$$\Omega(\mathbf{c}) + \Omega'(\mathbf{c})\Delta\mathbf{c} = 0.$$

Thus, when we define $\mathbf{b} \stackrel{\triangle}{=} -\Omega(\mathbf{c})$, and $\mathbf{G} \stackrel{\triangle}{=} \Omega'(\mathbf{c})$, we have a linearized **system state equation**, or the **state constraint equation**:

$$\mathbf{b} = \Omega'(\mathbf{c})\Delta\mathbf{c} = \mathbf{G}\mathbf{x}, \tag{7}$$

where $\mathbf{x} \in \mathbb{R}^r$ and $\mathbf{G} \in \mathbb{R}^{s \times r}$, s is the number of linear constraints. We have the following theorem that stipulates the kernel observability,

Theorem 1 Kernel-Observability
The system described by Eq. 7 and Eq. 6 is observable, i.e., unique recovery of $\mathbf{x}$ *is guaranteed, iff*

$$\texttt{rank}(\mathbf{M}^T\mathbf{M} + \lambda\mathbf{G}^T\mathbf{G}) = r, \qquad \forall\lambda > 0 \tag{8}$$

i.e., $(\mathbf{M}^T\mathbf{M} + \lambda\mathbf{G}^T\mathbf{G})$ *is of full rank.*

Proof: Please see Appendix.

Based on this theorem, we demonstrate three examples on the unique recovery of $\mathbf{x}$, i.e., $\Delta\mathbf{c}$, from the above system, and motivate our proposed approach of multiple collaborative kernels in Section 4.

3.1. Example 1: A Single Kernel

As a special case, if we do not consider the system state equation, which means the contribution of $\mathbf{G}$ vanishes, i.e., $\mathbf{G} = 0$, the observability of a single kernel, based on the Theorem, is given by checking $\texttt{rank}(\mathbf{M}^T\mathbf{M})$, or $\texttt{rank}(\mathbf{M})$[1].

This conclusion coincides with the SSD-based analysis in [7], where a least square problem is formulated:

$$\min_{\Delta c} \|\sqrt{\mathbf{q}} - \sqrt{\mathbf{p}(\mathbf{c})} - \frac{1}{2}\texttt{d}(\mathbf{p}(\mathbf{c}))^{-\frac{1}{2}}\mathbf{U}^T\mathbf{J}_\mathbf{K}(\mathbf{c})\Delta\mathbf{c}\|^2.$$

Hager *et.al.* [7] pointed out the rank deficiency of $\mathbf{M} = \frac{1}{2}\texttt{d}(\mathbf{p})^{-\frac{1}{2}}\mathbf{U}^T\mathbf{J}_\mathbf{K}(\mathbf{c})$ will not allow a unique solution to $\Delta\mathbf{c}$.

In order to recover $\Delta\mathbf{c}$ in this system, before taking effort to make $\mathbf{M}$ full rank, it should be noted that $\texttt{d}(\mathbf{p})^{-\frac{1}{2}}$ and $\mathbf{U}$ would not be rank deficient as long as the number of the non-zero values in the histogram is no less than the number of the parameters to be estimated, which is solely determined by the image and the target property.

Thus, the point that the artificial intervention can found its place is to change the kernel related $\mathbf{J}_\mathbf{K}(\mathbf{c})$, i.e., to change the ways of extracting the representative information from the objects, which motivates the methods of using multiple kernels. Two examples will be given in Sec 3.2 and Sec. 3.3, and our proposed method in Sec. 4.

3.2. Example 2: Kernel Concatenation

We can concatenate multiple kernels to increase the dimensionality of the measurement (i.e., the histogram). Suppose there are w kernels, each of them produces a histogram measure for the object, $\mathbf{p}_i(\mathbf{c}) = \mathbf{U}^T\mathbf{K}_i(\mathbf{c})$, where $i = 1, \ldots, w$. By vertically stacking these histograms into $\overline{\mathbf{p}}$ and $\overline{\mathbf{q}}$, it is easy to show that based on the Theorem, the observability is given by checking $\texttt{rank}(\mathbf{M}^T\mathbf{M})$, where

$$\mathbf{M} = \frac{1}{2}\texttt{d}(\overline{\mathbf{p}})^{-\frac{1}{2}} \begin{bmatrix} \mathbf{U}^T & & \\ & \ddots & \\ & & \mathbf{U}^T \end{bmatrix} \begin{bmatrix} \mathbf{J}_{\mathbf{K}_1} \\ \vdots \\ \mathbf{J}_{\mathbf{K}_w} \end{bmatrix}, \tag{9}$$

[1] For matrix $\mathbf{H}$, $\texttt{rank}(\mathbf{H}^T\mathbf{H}) = \texttt{rank}(\mathbf{H})$

which may hopefully have full column rank to enable a unique solution to $\Delta\mathbf{c}$. This makes sense since more features have been used. This is actually the multiple kernel method suggested in [7]. In fact, the kernel concatenation implies the optimization problem as:

$$\min_{\Delta c} \sum_{i=1}^{w} \|\sqrt{\mathbf{q}} - \sqrt{\mathbf{p}_i(\mathbf{c}+\Delta\mathbf{c})}\|^2.$$

3.3. Example 3: Kernel Combination

Besides kernel concatenation in Sec. 3.2 that uses more features, another feasible solution is kernel combination to produce new features by aggregating the histogram vectors from multiple kernels (with normalization):

$$\overline{\mathbf{q}} = \sum_{i=1}^{w} \mathbf{U}^T\mathbf{K}_i, \qquad \overline{\mathbf{p}} = \sum_{i=1}^{w} \mathbf{U}^T\mathbf{K}_i(\mathbf{c}).$$

Then the measurement equation is written as:

$$\sqrt{\overline{\mathbf{q}}} - \sqrt{\overline{\mathbf{p}}(\mathbf{c})} = \mathbf{M}\Delta\mathbf{c},$$

where

$$\mathbf{M} = \frac{1}{2}\mathrm{d}(\overline{\mathbf{p}})^{-\frac{1}{2}}\mathbf{U}^T\sum_{i=1}^{w}\mathbf{J}_{\mathbf{K}_i}. \qquad (10)$$

This may also make the matrix $\mathbf{M}^T\mathbf{M}$ full rank. In essence, as long as the measurement matrix $\mathbf{M}$ can well depict the characteristics around $\mathbf{c}$, we can find a proper $\Delta\mathbf{c}$ in the neighborhood that minimizes $\sqrt{\overline{\mathbf{q}}} - \sqrt{\overline{\mathbf{p}}(\mathbf{c}+\Delta\mathbf{c})}$.

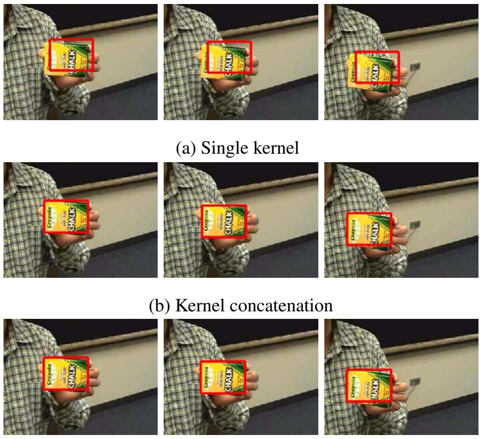
Here, we give an illustrative example. For comparison, we employ the same roof kernels as in [7], with length l, span s, center $\mathbf{c}$ and normal vector $\mathbf{n}$.

$$K_{roof}(\mathbf{x};\mathbf{c},\mathbf{n}) = \frac{4}{(l * s^2)}\max(\frac{s}{2} - \|(\mathbf{x}-\mathbf{c})\cdot\mathbf{n}\|, 0).$$

Intuitively, this is a truncated triangular kernel with preferred orientation $\mathbf{n}$. We implement a single roof kernel, a concatenation of two roof kernels with orthogonal orientations as Eq. 9 and a combination of the same two roof kernels as Eq. 10 to track a chalk box as shown in Fig. 1. The surfaces of value $1 - \|\sqrt{\mathbf{q}} - \sqrt{\mathbf{p}}\|^2$ w.r.t $\Delta\mathbf{c}$ generated by these three methods in one of the tracking iterations are plotted in Fig. 2.

It is clear that because of the rank deficiency, a single kernel cannot perceive the changes of $\Delta\mathbf{c}$ in some specific directions. While concatenated or combined kernels can well approximate the neighborhood values and ensure to find the $\Delta\mathbf{c}$ minimizing the error $\|\sqrt{\mathbf{q}} - \sqrt{\mathbf{p}}\|$, (in Figure 2, that is maximizing $1 - \|\sqrt{\mathbf{q}} - \sqrt{\mathbf{p}}\|^2$). Concatenated kernels and combined kernels have shown similarly better performance.

However, although these two multiple kernel methods may outperform the single kernel method, neither of them provides a principled way of designing multiple kernels.

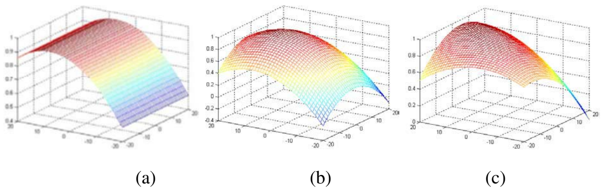


(a) Single kernel

(b) Kernel concatenation

(c) Kernel combination

Figure 1. A comparison of single kernel (top row), kernel concatenation (middle row), and kernel combination (bottom row).



(a)

(b)

(c)

Figure 2. The surfaces of $1 - \|\sqrt{\mathbf{q}} - \sqrt{\mathbf{p}}\|^2$ of (a) single kernel, (b) kernel concatenation, and (c) kernel combination.

4. Multiple Collaborative Kernels

As shown by the examples in Sec. 3, it is clear that we expect better performance than single kernel methods by using multiple kernels in the measurement process (e.g., Eq. 6). Based on the kernel observability Theorem, we notice that most existing multiple kernel methods [3, 7], including kernel concatenation and kernel combination, do not utilize the state constraints (i.e., Eq. 7), which should also be used to cope with rank deficiency. The neglect of the state constraints will largely limit the applicability of these methods, especially for complex motions. This is also one reason that holds the kernel methods back from tracking multiple targets, since simply assigning independent kernels on multiple targets is unlikely to solve the problem.

A new scheme, *multiple collaborative kernels*, is proposed in this section by exploiting the state equation $\Omega(\mathbf{x}) = 0$. We show that this is also an efficient way to improve the "observability" of the tracking system (Sec. 4.1). Our analysis also reveals the "collaboration" of multiple kernels that makes possible efficient computation (Sec. 4.2).

4.1. Enhancing the Observability

To start with, an obvious and commonly encountered prototype of $\Omega(\mathbf{x}) = 0$ for multiple targets would be the *structural constraint*. Taking a rigid rod as a simple example, see Fig. 3, we now show the improved "kernel-observability".

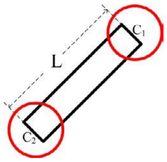


Figure 3. The length constraint on a rod.

Considering the slim shape of the rod, it is difficult to track it with one simple symmetric kernel. Alternatively, we can relax its motion by representing it as the joint motion of the two ends, while enforcing the length constraint on this relaxed (higher-dimensional) motion. Two simple symmetric kernels take charge of the two ends respectively. The benefit of doing this, besides recovering the rod position, is the estimation of the rod orientation.

By exploring the structural constraint, say, the rod is of fixed length L, we have[2],

$$\|\mathbf{c}_1 - \mathbf{c}_2\|^2 = L^2, \tag{11}$$

where, $\mathbf{c}_1$ and $\mathbf{c}_2$ are the resulting centers of the kernels placed at the ends. Then the objective function, which jointly considers both of the two kernels, will be formulated as:

$$O(\mathbf{c}_1, \mathbf{c}_2) = \sum_{i=1}^{2} \|\sqrt{\mathbf{q}_i} - \sqrt{\mathbf{p}_i(\mathbf{c}_i)}\|^2 + \gamma\|L^2 - \|\mathbf{c}_1 - \mathbf{c}_2\|^2\|^2,$$

where $\mathbf{q}_1$, $\mathbf{p}_1(\mathbf{c}_1)$ are the target model and the measured candidate associated with one of the ends, similarly with $\mathbf{q}_2$ and $\mathbf{p}_2(\mathbf{c}_2)$. This formulation compromises the *feature similarities* and the *structural constraint*, with γ being the tradeoff. By linearizing it at $(\mathbf{c}_1, \mathbf{c}_2)$, we have a linear system (with state equation and measurement equation):

$$\begin{cases} l = \mathbf{G}\begin{bmatrix}\Delta\mathbf{c}_1 \\ \Delta\mathbf{c}_2\end{bmatrix} \\ \mathbf{y} = \mathbf{M}\begin{bmatrix}\Delta\mathbf{c}_1 \\ \Delta\mathbf{c}_2\end{bmatrix} \end{cases}, \tag{12}$$

where

$$\overline{\mathbf{q}} = \begin{bmatrix}\mathbf{q}_1 \\ \mathbf{q}_2\end{bmatrix}, \overline{\mathbf{p}} = \begin{bmatrix}\mathbf{p}(\mathbf{c}_1) \\ \mathbf{p}(\mathbf{c}_2)\end{bmatrix}, \mathbf{y} = \begin{bmatrix}\sqrt{\mathbf{q}_1} - \sqrt{\mathbf{p}_1(\mathbf{c}_1)} \\ \sqrt{\mathbf{q}_2} - \sqrt{\mathbf{p}_2(\mathbf{c}_2)}\end{bmatrix},$$

$$\mathbf{M} = \begin{bmatrix}\mathbf{M}_1 & 0 \\ 0 & \mathbf{M}_2\end{bmatrix},$$

$$\mathbf{M}_i = \tfrac{1}{2}\mathtt{diag}(\mathbf{p}(\mathbf{c}_i))^{-\frac{1}{2}}\mathbf{U}_i^T\mathbf{J}_\mathbf{K}(\mathbf{c}_i), \quad i = 1, 2$$

$$\mathbf{G} = 2\left[(\mathbf{c}_1 - \mathbf{c}_2)^T \quad (\mathbf{c}_2 - \mathbf{c}_1)^T\right],$$

$$l = L^2 - \|\mathbf{c}_1 - \mathbf{c}_2\|^2.$$

Based on the kernel observability Theorem in Sec. 3, the observability of this formulation is given by checking $\mathtt{rank}(\mathbf{M}^T\mathbf{M} + \gamma\mathbf{G}^T\mathbf{G})$. This is equivalent to the column rank of $\begin{bmatrix}\mathbf{M} \\ \sqrt{\gamma}\mathbf{G}\end{bmatrix}$, which will be no less than that of $\mathbf{M}$.

Then, we can generalize the above idea by considering multiple kernels with a certain structural constraint $\Omega(\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_w) = 0$. The objective function will thus have the form,

$$\begin{aligned} O(\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_w) &= \sum_{i=1}^{w} \|\sqrt{\mathbf{q}_i} - \sqrt{\mathbf{p}_i(\mathbf{c}_i)}\|^2 \\ &+\gamma\|\Omega(\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_w)\|^2. \end{aligned} \tag{13}$$

After the linearization w.r.t. $\Delta\mathbf{c}_1, \Delta\mathbf{c}_2, \ldots, \Delta\mathbf{c}_w$, we have the following general system state equation and measurement equation:

$$\begin{cases} l = \mathbf{G}\Delta\overline{\mathbf{c}} \\ \mathbf{y} = \mathbf{M}\Delta\overline{\mathbf{c}} \end{cases}, \tag{14}$$

where

$$\Delta\overline{\mathbf{c}} = \begin{bmatrix}\Delta\mathbf{c}_1 \\ \Delta\mathbf{c}_2 \\ \cdots \\ \Delta\mathbf{c}_w\end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix}\sqrt{\mathbf{q}_1} - \sqrt{\mathbf{p}(\mathbf{c}_1)} \\ \sqrt{\mathbf{q}_2} - \sqrt{\mathbf{p}(\mathbf{c}_2)} \\ \cdots \\ \sqrt{\mathbf{q}_w} - \sqrt{\mathbf{p}(\mathbf{c}_w)}\end{bmatrix},$$

$$\mathbf{M} = \begin{bmatrix}\mathbf{M}_1 & 0 & 0 & 0 \\ 0 & \mathbf{M}_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \mathbf{M}_w\end{bmatrix}, \tag{15}$$

$$\mathbf{G} = \begin{bmatrix}\frac{\partial\Omega}{\partial\mathbf{c}_1} & \frac{\partial\Omega}{\partial\mathbf{c}_2} & \cdots & \frac{\partial\Omega}{\partial\mathbf{c}_w}\end{bmatrix},$$

$$l = -\Omega(\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_w).$$

Similarly, the unique motion can be estimated, provided that $\begin{bmatrix}\mathbf{M} \\ \sqrt{\gamma}\mathbf{G}\end{bmatrix}$ has full column rank.

Now, it is worth pointing out that without the introduced constraint $\Omega(\cdot)$, i.e., $\mathbf{G} = \mathbf{0}$, the solution will be reduced to

$$\mathbf{y} = \mathbf{M}\Delta\overline{\mathbf{c}}, \tag{16}$$

which is equivalent to solving the w kernel tracking problems *independently*, requiring $\mathbf{M}$ to have full column rank, i.e., every kernel needs to be observable.

The advantage of the collaborative kernels is that it does not require all the kernels to be fully observable. Even if some of the kernels get bad, e.g., distracted by the clutters, the other kernels may still be able to "pull" the ill-behaved kernels back to the track according to the inherent constraint embedded in Eq. 14. As long as $(\mathbf{M}^T\mathbf{M} + \gamma\mathbf{G}^T\mathbf{G})$ is full rank, our method can tolerate those unobservable kernels. In theory, such a good property is guaranteed by the fact that $\mathtt{rank}(\begin{bmatrix}\mathbf{M} \\ \sqrt{\gamma}\mathbf{G}\end{bmatrix}) \geq \mathtt{rank}(\mathbf{M})$.

[2] This simple constraint is for illustrative purpose. More complex constraint will be readily incorporated.

More importantly, notice that the structural constraint is just one of the prototypes of the system description, $\Omega(\mathbf{x}) = 0$, the above paradigm of design for multiple collaborative kernels can be readily extended to other system descriptions with different physical meanings, such as the more complicated motion dynamics or the learned motion priors.

4.2. The Collaboration

The solution to the linear system in our formulation (Eq. 14) for multiple collaborative kernel tracking is given by:

$$\Delta\overline{\mathbf{c}} = (\mathbf{M}^T\mathbf{M} + \gamma\mathbf{G}^T\mathbf{G})^{-1}(\mathbf{M}^T\mathbf{y} + \gamma\mathbf{G}^T l). \qquad (17)$$

Due to the relaxation of the system states, the dimension of the matrix $(\mathbf{M}^T\mathbf{M} + \gamma\mathbf{G}^T\mathbf{G})$ can be quite large (the sum of motion parameters of all individual kernels). Thus, it is computationally demanding to calculate its inverse. Considering the special structure of $\mathbf{M}$, we obtain a much more efficient method, which amazingly reveals the collaboration among multiple kernels.

By applying matrix inversion lemma[3], we can obtain,

$$\Delta\overline{\mathbf{c}} = (\mathbf{I} - \mathbf{D})(\mathbf{M}^T\mathbf{M})^{-1}(\mathbf{M}^T\mathbf{y} + \gamma\mathbf{G}^T l), \qquad (18)$$

where $\mathbf{D} = \gamma(\mathbf{M}^T\mathbf{M})^{-1}\mathbf{G}^T(\gamma\mathbf{G}(\mathbf{M}^T\mathbf{M})^{-1}\mathbf{G}^T + \mathbf{I})^{-1}\mathbf{G}$

Providing that $\mathbf{M}^T\mathbf{M}$ is non-singular, this equation means that we can save the computational cost on $(\mathbf{M}^T\mathbf{M} + \gamma\mathbf{G}^T\mathbf{G})^{-1}$ by computing $(\gamma\mathbf{G}(\mathbf{M}^T\mathbf{M})^{-1}\mathbf{G}^T + \mathbf{I})^{-1}$ and $(\mathbf{M}^T\mathbf{M})^{-1}$ instead. Generally, the dimensionality of $(\gamma\mathbf{G}(\mathbf{M}^T\mathbf{M})^{-1}\mathbf{G}^T + \mathbf{I})$, which equals the number of constraint, is smaller than the parameters to be estimated, i.e., the dimensionality of $(\mathbf{M}^T\mathbf{M} + \gamma\mathbf{G}^T\mathbf{G})$. Moreover, the calculation of $(\mathbf{M}^T\mathbf{M})^{-1}$ is not difficult since it has a block-diagonal structure form (recalling the structure of $\mathbf{M}$ in Eq. 15). All of these count to a potential decrease in the computational cost.

Noticing that the solution to the unconstrained problem (i.e., independent kernels) is given by:

$$\Delta\overline{\mathbf{c}}_u = (\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T\mathbf{y} = \mathbf{M}^\dagger\mathbf{y}, \qquad (19)$$

where $\mathbf{M}^\dagger$ is the pseudo-inverse of $\mathbf{M}$. This unconstrained solution can be calculated easily with linear cost w.r.t. the number of kernels, since $\mathbf{M}$ is a block diagonal matrix. Any single kernel tracking method can be applied here.

Using the unconstrained solution $\Delta\overline{\mathbf{c}}_u$, we can rewrite the solution to the constrained problem (i.e.,Eq. 18) as:

$$\Delta\overline{\mathbf{c}} = (\mathbf{I} - \mathbf{D})\Delta\overline{\mathbf{c}}_u + \mathbf{z}(\mathbf{c}), \qquad (20)$$

where $\mathbf{z}(\mathbf{c}) = \gamma(\mathbf{I} - \mathbf{D})(\mathbf{M}^T\mathbf{M})^{-1}\mathbf{G}^T l$. The mechanism of the collaboration among multiple kernels is pronounced: each individual single-kernel tracker follows its designated target (a small part of the entire target of interest) by its own means, and exchanges "corrections" to other single-kernel tracker. Such a collaboration ends up with an equilibrium where the entire target is tracked and the structural constraints among multiple kernels are satisfied.

The collaboration actually suggests a very efficient recursive method of calculating the constrained solution. We can alternate two steps until convergence: first relax the constraints to solve the unconstrained one by Eq. 19, and then adjust the unconstrained estimates according to Eq. 20, with less computational cost.

$$\Delta\overline{\mathbf{c}}^{k+1} \longleftarrow (\mathbf{I} - \mathbf{D}^k)[\mathbf{M}(\Delta\overline{\mathbf{c}}^k)]^\dagger\mathbf{y}^k + \mathbf{z}^k, \qquad (21)$$

which is very similar to the fixed point iteration and converges very fast.

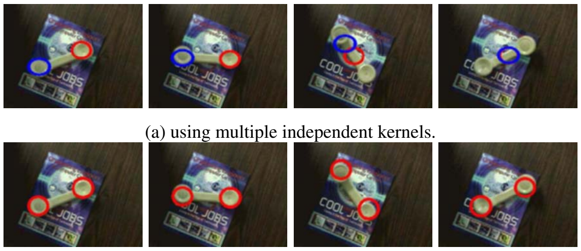
This collaborative solution is very useful to multiple target tracking. Because we avoid estimating the motion states from the joint parameter space. Instead, we solve the divided problems in the reduced solution space, then applying regularized terms to meet the certain constraint.

5. Experiments

In this section, we report our experiments of the proposed multiple collaborative kernel method to track structured objects and articulated objects, and the comparison to multiple independent kernel tracker.

5.1. Tracking structured object

Object with certain spatial structure is a commonplace in many tracking tasks. But some of them, such as a handset or a rod-shaped bottle, cannot be easily handled by the tracker with a single symmetric kernel. See Fig. 4 and Fig. 5. Our experiments validate the proposed method of multiple collaborative kernels that can track these targets successfully and estimate the target orientation as a byproduct.


(a) using multiple independent kernels.

(b) using multiple collaborative kernels.

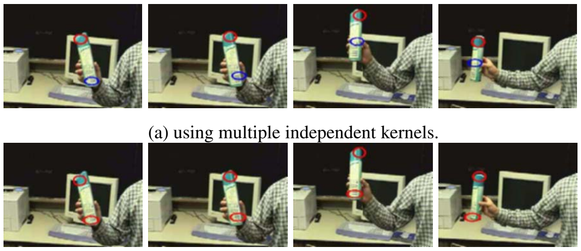
Figure 4. Tracking a handset.

Fig. 4 shows 4 sample frames from a sequence of a rotating handset. The histogram in the RGB space is taken as the feature. We first apply two independent normal kernels at both ends of the handset, colored as red and blue, respectively. The result is shown in Fig. 4(a). We should

[3] $(\mathbf{A} + \mathbf{BD})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}(\mathbf{DA}^{-1}\mathbf{B} + \mathbf{I})^{-1}\mathbf{DA}^{-1}$, where **A** is a n by n matrix, **B** is a n by m matrix and **D** is a m by n matrix

notice that the motion along the handset is not fully observable for both kernels, and the appearances of the two ends of the handset are identical. The two kernels drift along the handset and eventually lose the track.

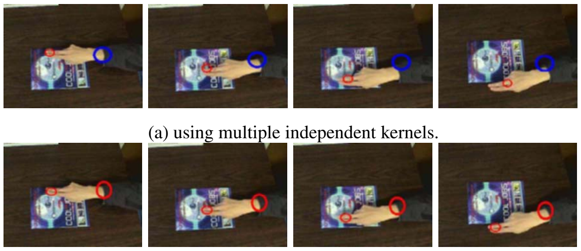
With the same kernels but collaborating them based on our method, we introduce a length constraint, $\|\mathbf{c}_1 - \mathbf{c}_2\|^2 = L^2$, with L given by the initialization. The result is shown in Fig. 4(b). As predicted, the collaboration of the two kernels leads to a successful tracking result. This experiment shows a quite meaningful property of the collaborative kernel approach: although not all the kernels are fully observable, the collaboration can still make the ensemble observable. In all experiments, we set γ in Eq. 13 to be 1.


(a) using multiple independent kernels.

(b) using multiple collaborative kernels.

Figure 5. Tracking a rod-shaped bottle.

Fig. 5 shows another experiment on a rod-shaped bottle. We first place two independent normal kernels at the ends. The histogram of H-value of the HSV space is used as the object feature. Sample frames of the result of using independent kernels are shown in Fig. 5(a). Notice that the motion of the lower-end, indicated by the kernel in blue, is not fully observable, since the image regions in the lower part of the bottle are similar. Thus the blue kernel is vulnerable to distraction when the two kernels function independently. In contrast, the collaboration of the two kernels contributes to a more reliable tracking result, as shown in Fig. 5(b).


(a) using multiple independent kernels.

(b) using multiple collaborative kernels.

Figure 6. Tracking a finger.

The proposed collaborative scheme also provides another benefit. When a certain target is our focus-of-attention but unfortunately cannot be stably tracked, we can refer to another easily tracked object as an auxiliary to gain a better result. In Fig. 6, we aim to track the fingertip in a clutter. By placing a kernel on the easily tracked wrist, we constrain the two kernels with a fixed length. The result of using our method is shown in Fig. 6(b). In fact, The roles of the object of attention and the auxiliary are interchangeable throughout the process in order to ameliorate the potential tracking failure of either one. Two independent kernels, as shown in Fig. 6(a), of course are unable to recover from tracking failure in the clutter.

5.2. Tracking articulated objects

Another useful application of multiple collaborative kernel tracking is to track articulated targets, such as human body articulation. To the best of our knowledge, this is the first work extending kernel methods into this task.

Fig. 7 shows sample frames of an experiment, in which a person moves his two arms. We apply two pairs of collaborative kernels on the elbows and the hands. The tracking result of our approach is shown in Fig. 7(b). On the contrary, the method based on four independent kernels leads to a much inferior performance, as shown in Fig. 7(a).

Another experiment on an articulated structure consisting of a arm and a bottle in hand is shown in Fig. 8. We apply three kernels to the elbow, the hand and one end of the bottle, respectively. Compared with the result yielded by independent kernels (in Fig. 8(a)), two pairs of collaborative kernels (elbow & hand, hand & bottle tip) provide a much more robust performance as shown in Fig. 8(b).

The structural constraint used here serves as a basic means facilitating the implementation of multiple collaborative kernels on more complex tracking tasks.

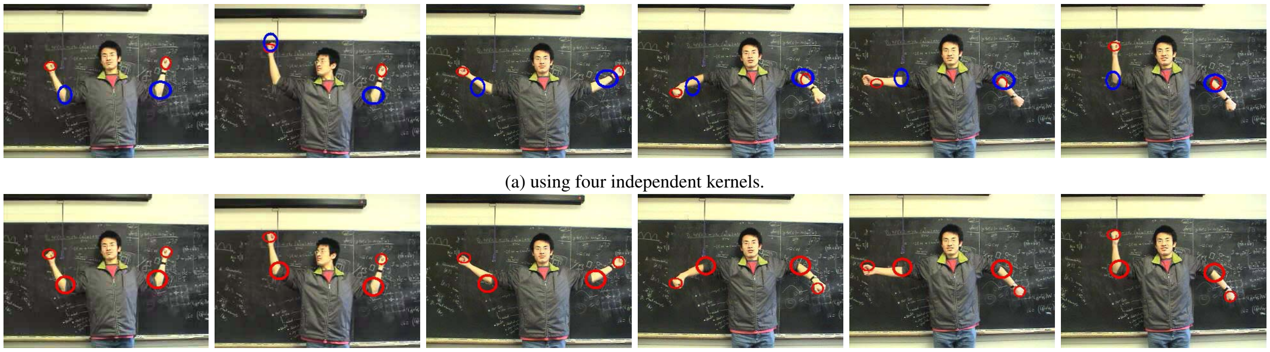
6. Conclusions

To summarize, in this paper, a criterion is obtained on the issue of “kernel-observability”, which leads to a principled way of kernel design with prevention of singularity in kernel based tracking problems. Based on this, a multiple collaborative kernel tracking scheme is proposed. Different from the most existing kernel based algorithms, which are confined by independent kernels and single target, we show that by exploiting the inherent relationship among multiple kernels, not only the “kernel-observability” is improved, but also the applicability of the kernel based methods is naturally extended to cope with articulated targets and complex motions. This helps to gain more insight into the role that kernel plays in the tracking problems.

However, we are using the geometric constraint in this paper to improve the kernel-observability, which is rigid in its current form. The focus of our future work will be exploring how to incorporate richer system models to account for more complicated motions and how to make the kernel design adaptable to various environmental changes.

Appendix

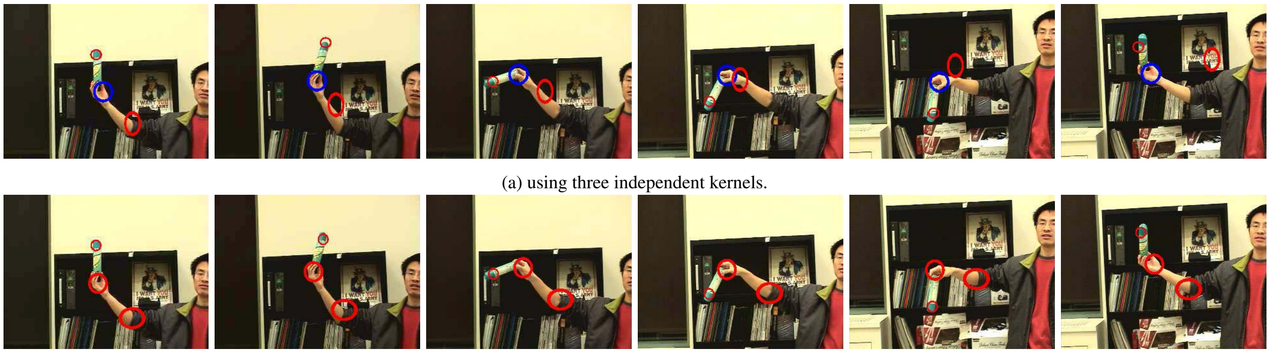
We give a brief proof of Theorem 1. Given the system state equation (Eq. 7) and the measurement equation (Eq. 6),



(a) using four independent kernels.

(b) using two pairs of collaborative kernels.

Figure 7. Tracking the articulated body with two arms.



(a) using three independent kernels.

(b) using two pairs of collaborative kernels.

Figure 8. Tracking an articulated structure.

we form an objective function that penalizes the measurement mismatch and the deviation from system constraints:

$$L(\mathbf{x}) \triangleq \|\mathbf{M}\mathbf{x} - \mathbf{y}\|^2 + \lambda\|\mathbf{G}\mathbf{x} - \mathbf{b}\|^2,$$

where $\lambda > 0$. Setting the derivative to zero, we have:

$$\mathbf{x}^* = (\mathbf{M}^T\mathbf{M} + \lambda\mathbf{G}^T\mathbf{G})^{-1}(\mathbf{M}^T\mathbf{y} + \lambda\mathbf{G}^T\mathbf{b}).$$

This is equivalent to the least square solution to the following system:

$$\begin{bmatrix} \mathbf{M} \\ \sqrt{\lambda}\mathbf{G} \end{bmatrix} \mathbf{x} = \begin{bmatrix} \mathbf{y} \\ \sqrt{\lambda}\mathbf{b} \end{bmatrix}.$$

Thus the solution is unique iff the rank of $\mathbf{M}^T\mathbf{M} + \lambda\mathbf{G}^T\mathbf{G}$ is full, or $\begin{bmatrix} \mathbf{M} \\ \sqrt{\lambda}\mathbf{G} \end{bmatrix}$ has full column rank. ■

Acknowledgement

This work was supported in part by National Science Foundation (NSF) Grant IIS-0308222, NSF IIS-0347877 (CAREER), Northwestern startup funds and the Murphy Fellowships.

References

[1] A. Blake and M. Isard. *Active Contours*. Springer-Verlag, London, 1998.

[2] Y. Cheng, "Mean Shift, Mode Seeking, and Clustering", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1995, vol. 17, no. 8, pp. 790-799.

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

[4] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, vol. 24, no. 5, pp. 603-619.

[5] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift", *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2000, vol. 2, pp. 142-149.

[6] D. Comaniciu, V. Ramesh, and P. Meer, "The variable bandwidth mean shift and data-driven scale selection", *Proc. International Conference on Computer Vision*, 2001, vol. 1, pp. 438-445.

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

[8] M. Isard and A. Blake, "CONDENSATION – conditional density propagation for visual tracking", *Int'l J. Computer Vision*, 1998, vol. 29, pp. 5-28.

[9] M. P. Wand and M. C. Jones, *Kernel Smoothing*, Chapman and Hall, 1995, First edition.

[10] J. Wang, B. Thiesson, Y. Xu, and M. F. Cohen, "Image and video segmentation by anisotropic kernel mean shift", *Proc. European Conference on Computer Vision*, 2004.

[11] Y. Wu, G. Hua, and T. Yu, "Tracking Articulated Body by Dynamic Markov Network", *Proc. International Conference on Computer Vision*, 2003, pp. 1094-1101.