FRAGMENT-BASED TRACKING USING ONLINE MULTIPLE KERNEL LEARNING

Xu Jia, Dong Wang, Huchuan Lu

School of Information and Communication Engineering
Dalian University of Technology, Dalian, CHINA

ABSTRACT

Fragment-based tracking methods have shown its robustness in handling partial occlusion and pose change. In this paper, we propose a novel fragment-based tracking approach using on online multiple kernel learning (MKL) method. An online MKL method for object tracking is implemented by considering temporal continuity explicitly. Instead of directly using multiple features of objects, we employ MKL to make full use of multiple fragments of the object. This can automatically assign different weights to the fragments according to their discriminative power. In addition, for better robustness two kinds of independent features are computed to enrich the representation of patches. We build a classifier for each type of feature and assign them different weights according to their performance on classification. Both qualitative and quantitative evaluations on challenging image sequences demonstrate that the proposed tracking approach performs favorably against several state-of-the-art methods.

***Index Terms*—** object tracking, multiple kernel learning (MKL), fragment-based tracking

1. INTRODUCTION

For designing a robust online model-free tracker, one of the most important issues is to develop an effective appearance model that can handle both intrinsic (e.g., pose and shape variations) and extrinsic (e.g., illumination change and partial occlusion) factors. From the perspective of object representation, appearance model can be either global-based or local-based. Tracking methods based on holistic representation (e.g., Meanshift [1] and IVT [2]) treat the object as an entity. Though these methods are computationally efficient, they tend to lose the object structure information and are sensitive to illumination change and partial occlusion. In contrast, trackers based on local representation (e.g., Co-Tracking [3], MIL [4] PN [5] and SPT [6]) usually model the tracked target as a collection of local features and fed them to classifiers to distinguish the target from background. They are effective in handling partial occlusion, however the high dimension of feature requires expensive computational costs for learning and testing. Besides, they often confuse the target with similar background due to the loss of global information.

Adam et al. [7] propose a fragment-based tracking method and further Wang et al. [8] embed the fragment-based representation into mean shift framework. It computes the voting maps of multiple fragments by comparing with the histograms of corresponding fragments within the template and then combine them to make estimation. However, equal importance being assigned to those patches and static template limit the performance of that tracker.

Recently, multiple kernel learning (MKL) method has been applied in object classification and recognition task, showing great advantages on the improvement of accuracy [9, 10]. MKL addresses the classification problem by learning and optimizing a multiple-kernel classifier objective from training data. Different weights of multiple features obtained can be used to measure their contribution to the overall discriminative power. MKL has been introduced into object tracking by Yang et al. [11] and achieves good performance. However, their method does not adopt fragment-based representation and hence may be sensitive to partial occlusion.

For visual tracking, to the best of our knowledge, there are few methods that work on combining multiple fragments with different weights. In this paper, we propose a fragment-based tracking method which use online multiple kernel learning (MKL) method to adaptively integrate the discriminative power of multiple fragments of the object. Similar to FragTrack [7], we also represent the object with histograms of multiple fragments. However, we take advantage of multiple fragments under a discriminative framework. We employ MKL classifier to find the optimal boundary which is able to separate the object from background. Different weights are assigned to multiple patches by MKL classifier to maximize the discriminative power. For better robustness, two types of complementary features are extracted to train independent MKL classifiers. Final estimation of the target is based on the combined decision of those classifiers. The classifiers are updated in a conservative way to reduce drifting problem.

Contributions The contributions of this paper include:

- We implement an online MKL method for object tracking which explicitly considers temporal continuity.
- A fragment-based tracking method based on multiple kernel learning (MKL) is proposed.
- Tracking performance is further improved by integrating two independent types of features.

2. ONLINE MULTIPLE KERNEL LEARNING FOR OBJECT TRACKING

Multiple kernel learning (MKL) is an extension of kernel learning methods (especially kernel SVM). By using different types of kernel to depict different properties of samples (e.g., feature and metric), MKL provides a unified framework for model combination and selection. One of the most influential works is the SimpleMKL method proposed by Rakotomamonjy *et al.* [12], which defines kernel function as a convex linear combination of kernels,

$$K\left(\mathbf{x},\mathbf{x}'\right)=\sum_{m=1}^{M}\beta_m K_m\left(\mathbf{x},\mathbf{x}'\right),\ \sum_{m=1}^{M}\beta_m=1,\beta_m\geq 0, \quad (1)$$

where $K_m\left(\mathbf{x},\mathbf{x}'\right)$ denotes the m-th kernel and β_m is the weight of each kernel. The SimpleMKL algorithm is aimed to simultaneously obtain support vectors, support vector coefficients and kernel weights by solving the following constrained optimization problem,

$$\min_{\beta} J\left(\beta\right)\ such\ that\ \sum_{m=1}^{M}\beta_m=1,\ \beta_m\geqslant 0, \quad (2)$$

where

$$J\left(\beta\right)=\begin{cases}\min\limits_{\{f\},b,\xi}\ \frac{1}{2}\sum\limits_{m}\frac{1}{\beta_m}\left\|f_m\right\|^2_{\mathcal{H}_m}+C\sum\limits_{i}\xi_i\ \forall i\\ s.t.\ y_i\sum\limits_{m}f_m\left(\mathbf{x}_i\right)+y_ib\geqslant 1-\xi_i\\ \xi_i\geqslant 0,\ \forall i.\end{cases}\tag{3}$$

In Eq. 3 $\mathbf{x}_i$ denotes i-th training sample, y_i and ξ_i represent its label and slack variable respectively and C is a penalty factor for slack variable. The SimpleMKL algorithm can be solved by two iterative steps: (1) fix β, it is reduced to be a standard SVM optimization problem; (2) fix $f(.)$, Rakotomamonjy *et al.* [12] solve β by using a reduced gradient method, which computes simple differentiation of the dual function of Eq. 3 with respect to β_m,

$$\frac{\partial J}{\partial\beta_m}=-\frac{1}{2}\sum_{i,j}\alpha_i\alpha_jy_iy_jK_m\left(\mathbf{x}_i,\mathbf{x}_j\right),\ \forall m,\tag{4}$$

where α_i stands for the dual coefficient of $\mathbf{x}_i$ (if $\alpha_i^*\neq 0$, $\mathbf{x}_i$ is also known as support vector). The obtained decision function of MKL classifier for binary classification can be written as

$$F_{MKL}\left(\mathbf{x}\right)=\sum_{i}\alpha_iy_i\sum_{m}\beta_mK_m\left(\mathbf{x},\mathbf{x}_i\right)+b.\tag{5}$$

However, the basic SimpleMKL method is not suitable for object tracking which requires online learning to adapt to the appearance change of both the target and background. In [13], an incremental MKL method is proposed for object recognition. We note that it is improper to directly extend it into object tracking since it does not take prior information (i.e., temporal continuity) of the tracking problem into consideration. In this study, we implement an online MKL method for object tracking that considers temporal continuity explicitly. The objective function of the t-th update is defined as

$$\min_{\beta^t}J\left(\beta^t\right)\ such\ that\ \sum_{m=1}^{M}\beta^t_m=1,\ \beta^t_m\geqslant 0,\tag{6}$$

where

$$J\left(\beta^t\right)=\begin{cases}\min\limits_{\{f,b_t,\xi\}}\ \frac{1}{2}\sum\limits_{m}\frac{1}{\beta^t_m}\left\|f^t_m\right\|^2_{\mathcal{H}_m}+C\sum\limits_{i}\xi_i\\ \qquad+\frac{1}{2}\lambda\sum\limits_{m}\left(\beta^t_m-\beta^{t-1}_m\right)^2,\ \forall i\\ s.t.\ y^t_i\sum\limits_{m}f_m\left(\mathbf{x}^t_i\right)+y^t_ib_t\geqslant 1-\xi_i\\ \xi_i\geqslant 0,\ \forall i.\end{cases}\tag{7}$$

The regularization term $\frac{1}{2}\lambda\sum\limits_{m}\left(\beta^t_m-\beta^{t-1}_m\right)^2$ models temporal continuity explicitly, where λ is a small constant. We can see that the form of our objective function (Eq. 6 and 7) is similar to that of the basic SimpleMKL algorithm (Eq. 2 and 3). Thus, we can use the SimpleMKL package [12] to solve our online MKL problem. Based on *Lagrangian dual theory*, the differentiation of the dual function of Eq. 7 with respect to β^t_m can be obtained as,

$$\frac{\partial J}{\partial\beta^t_m}=-\lambda\left(\beta^t_m-\beta^{t-1}_m\right)-\frac{1}{2}\sum_{i,j}\alpha^t_i\alpha^t_jy^t_iy^t_jK_m\left(\mathbf{x}^t_i,\mathbf{x}^t_j\right).\tag{8}$$

Due to space limitation, we merely highlight the difference of our online MKL method compared with SimpleMKL [12].

(1) Training Samples: To achieve online learning, we composite the training set by $\mathcal{X}^t=\left\{\mathcal{X}^{t-1}_{\text{sup}},\mathcal{X}^t_{new}\right\}$, where $\mathcal{X}^t=\left\{\mathbf{x}^t_1,\mathbf{x}^t_2,...\right\}$ denote the training samples at t-th update, $\mathcal{X}^{t-1}_{\text{sup}}$ stands for support vectors obtained by last update and $\mathcal{X}^t_{new}$ are new collected samples (The way to collect training samples is described in Section 3). This prevents the classifier from varying too abruptly.

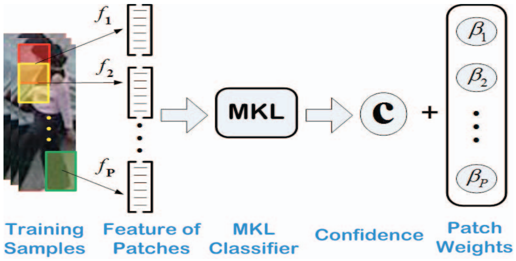


Fig. 1. Illustration about how we make use of MKL in the proposed fragment-based tracking.

(2) Temporal Continuity: Compared with SimpleMKL [12], we consider the temporal continuity prior explicitly by introducing an additional regularization term. The differentiation form for the reduced gradient method is derived as Eq. 8. So it is easy to implement the online MKL method by using the SimpleMKL package [12]. We note that the consideration of temporal continuity makes our implemented online MKL method more suitable for visual tracking.

3. FRAGMENT-BASED TRACKING USING MULTIPLE KERNEL LEARNING

Different from the tracker proposed by Adam et al. [7], we present an online discriminative framework to make full use of multiple fragments of object. Our method works by treating patches differently according to their discriminative power. Just like most tracking-by-detection methods [4, 5], we also adopt the output margin of classifier as the observation likelihood of the target.

3.1. Fragment-based Tracking Using MKL

Due to the success of HOG [14] in object detection and tracking tasks, we adopt the similar way about block division used in that paper to generate fragments. One feature vector is computed for each fragment and feature vectors of all fragments altogether convey structural and global information. Our goal is to find a strategy to integrate this information to maximize the overall discriminative power. MKL has shown its potential in integrating multiple features in recent research. But most of their inputs are concatenated huge feature vectors of high dimension and much redundant and confusing information are mixed into it. Instead, we deem each local patch as one distinctive feature of the object. Therefore, as for our problem, the output margin of MKL classifier can be re-written as follows:

$$F_{MKL}(x)=\sum_{i=1}^{N}\alpha_iy_i\sum_{p=1}^{P}\beta_pK_p(x,x_i)+b\tag{9}$$

where $K_p(x,x')=K(f_p(x),f_p(x'))$, $f_p(x)$ denotes the mapping function to feature space of patch p, parameter P denotes the total number of patches and β_p weighs the importance of each patch. Fig. 1 gives a simple illustration about how we make use of fragments and MKL classifier in our fragment-based tracking. This problem can be efficiently solved by the above mentioned online MKL (Section 2). The weights β_p of patches allow us to account for the fact that some patches within the window are more representative and discriminative, while others only contain redundant and confusing information. The weighted sum of these patch kernels preserve high discriminative performance. When the target is partially occluded or experiences small pose variation, the proposed method is able to pick out the most discriminative patches to separate the object from background (shown in Section 4.1).

3.2. Multi-cue combination and model update

Cues combination: Combination of independent features is able to complementarily enrich the representation of an object and improve robustness of tracking [3, 15]. Though kernel alignment has been proposed in [16], it is not quite proper to directly combine kernels of heterogeneous features, especially as for the small-sample problem like tracking. We build a classifier for each type of feature and assign them different weights with respect to their performance on classification. The weight of classifier $w_i, i = 1, 2$, is computed by

$$w_i = \begin{cases} \dfrac{\max\{c_i'\}}{\max\{c_1'\} + \max\{c_2'\}} & a_1, a_2 \geqslant T \\ \dfrac{a_i}{a_1 + a_2} & otherwise \end{cases} \tag{10}$$

where c_1' and c_2' denote confidence computed with the output of both classifiers in the frame prior to update, a_1 and a_2 represent the classification accuracy of each classifier on previous training set and T is a threshold on classification accuracy (T is set to 0.9 in this paper). Then final confidence c is the weighted sum of confidence c_1 and c_2 which are computed in the current frame by classifiers.

$$l = w_1 \times c_1 + w_2 \times c_2 \tag{11}$$

This means only under the condition that both classifiers are reliable, do weights depend on the ratio of classifiers' largest confidence. Otherwise, classification accuracy determines the weights of those two types of features. Since our tracking framework is based on the discriminative power of patches, the essence of multi-cue fusion here lies in that it finds the most suitable matching scheme for patch and feature type.

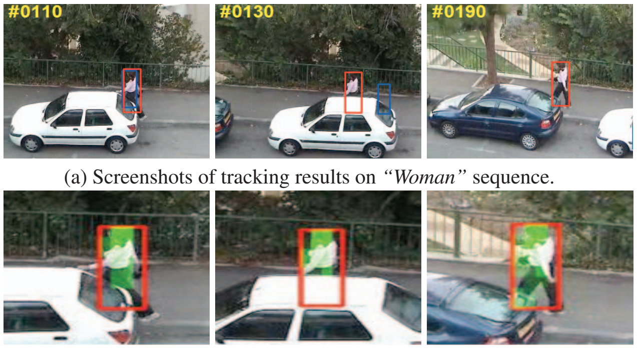
Sample collection and model update: To achieve online learning for MKL classifier, we collect new training samples in two ways, i.e. $\mathcal{X}^t = \{\mathcal{X}_{\text{sup}}^{t-1}, \mathcal{X}_{new}^t\}$. The positive samples of $\mathcal{X}_{new}^t$ are previously tracked results $\{X_{prev}\}$. A hard threshold on final confidence is set to prevent bad tracked results from being updated into training set. We experimentally choose a threshold value of 0.4 to compare with the maximum of confidence. This allows classifiers to update conservatively over time when it is not quite sure of the response. For negative ones, we crop out a set of samples around X_{prev} in four directions (up, down, left, right), which do not overlap with X_{prev}. $\{\mathcal{X}_{\text{sup}}^{t-1}\}$ denotes valid support vectors inherited from previously trained classifiers. This prevents the decision boundary from varying too abruptly to lead to drifting problem. Then we accumulate these samples up to B frames and send them as shared training data to update both classifiers.

4. EXPERIMENTS AND RESULTS

We implemented our tracker in MATLAB and tested it on several challenging image sequences, one from [7] and three from our own dataset. These challenges include partial occlusion, pose change, illumination variation and background clutter. The color feature used here is 27 ($3 \times 3 \times 3$) dimension RGB histogram; the HOG feature is computed with 9 orientation bins. For each individual feature, the histogram intersection kernel [17] is computed due to its simplicity and robustness. For our MKL classifier, we fix $C = 100$, $\lambda = 0.1$ and the parameter B is 5. The number of sampling particles is all set to 500 in this study. For simplification, we denote our fragment-based online multiple kernel learning method by FMKL.

4.1. The effectiveness of our FMKL tracking method

Adaptive weight kernel *vs* Average kernel: We note that our FMKL method learns adaptive weights for each fragment on the fly, in order to enhance the discriminative power of the tracker. To demonstrate its effectiveness, we compare it with the method using average kernel that assigns equal weights to different fragments (both methods only use color histogram as feature in this experiment). Fig. 2 shows some representative tracking results on *"Woman"* sequence. From Fig. 2 (a) we can see that the FMKL method performs better than the method using average kernel. Fig. 2 (b) shows the discriminative power of multiple fragments by visualizing their weights. The deeper green color of one patch is, the more it contributes to the overall discriminative power. It demonstrates that our FMKL method is able to identify important patches during the tracking process, thereby accurately estimating the location of the tracked target. In contrast, the method using average kernel treats each fragment equally. Thus, it may confuse the tracker especially when occlusion occurs (e.g., Fig. 2 #0130) and cause tracking drift.



(a) Screenshots of tracking results on *"Woman"* sequence.

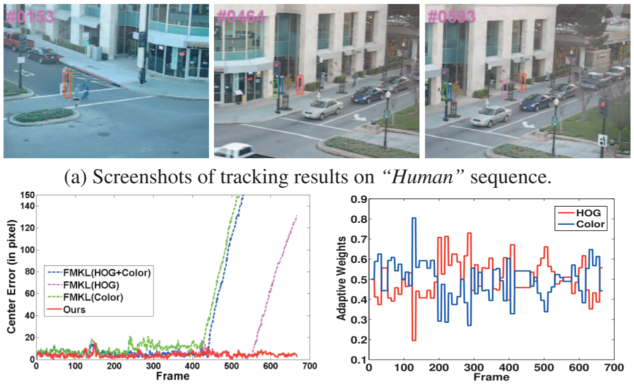
(b) Some examples that are cropped out to visualize the weights of patches.

Fig. 2. Comparison between adaptive weights kernel (red box) and average kernel (blue box) using "Woman" sequence.

The effectiveness of our cues combination strategy: Fig. 3 illustrates the effectiveness of our cues combination strategy (presented in Section 3.2) by using "Human" sequence. We have the following two observations: (1) the two cues FMKL method using our cues combination strategy performs better than the FMKL method using individual features since our cues combination strategy is able to adjust the weights of individual cues according to their performance (Fig. 3 (c)). RGB feature weighs more when there is obvious difference between the target and background while HOG feature that characterize shape information is more discriminative when background is of similar color to the target; (2) we also use MKL to directly fuse two independent features by using the kernel alignment strategy [16]. However, it does not work well (as shown in Fig. 3 (b) and (c)). We note that it is because the kernel alignment strategy is not suitable for the tracking problem. This also demonstrates the effectiveness of our cues combination strategy.

4.2. Comparison with state-of-the-art methods

The proposed method is compared with some state-of-the-art methods, including IVT [2], FragTrack [7], MIL [4], VTD [18] and PN [5]. We only give representative results here due to space limitation and more results are available on the website http://www.youtube.com/watch?v=U9FZc0B3lpE&feature=youtu.be. The qualitative and quantitative tracking results are respectively shown in Fig. 4 and Table 1. From the tracking results, we can see the proposed method performs favorably against state-of-the-art methods. In the first row of Fig. 4, a woman undergoes long-time partial occlusion and small pose variation. Our method could successfully handle these challenges because it picks out more informative patches and makes full use of them to discriminate the target from background. The PN tracker is able to re-acquire the target when the target object reappears after occlusion. However, other methods lock on a car of similar color to the woman's trousers. In the *"Human"* sequence, most of methods fail to track the man when he is occluded by a board. Though PN tracker can re-acquire



(a) Screenshots of tracking results on *"Human"* sequence.

(b) Quantitative Comparison

(c) Adaptive weights of cues

Fig. 3. The effectiveness of our cues combination strategy. This figure demonstrates the performance of the proposed FMKL method and cues combination strategy. Our tracking framework (presented in Section 3) is compared with FMKL methods using indiviual features (FMKL(HOG) and FMKL(Color)) and a FMKL method that directly fuse HOG and RGB features (denoted as FMKL(HOG+Color)).

the target after the he passes the board, it drifts to a woman of similar color clothes to the man. In the third row, the Frag tracker, PN tracker and ours can keep track of the target attributed to the handle of partial occlusion. While others fail when large occlusion occurs. In the *"Bolt"* sequence, most of methods fail to track the player which is of much pose variation and background clutter. However, our tracker is able to track the target because it is able to pick out the most informative and discriminative patches. The optimized matching scheme of multiple fragments and multiple cues helps locate the player.

	IVT	Frag	MIL	VTD	PN	Ours
Woman	167.5	138.1	122.4	136.6	9.0	**3.2**
Human	191.0	211.3	174.5	182.7	-	**4.1**
Girl	44.2	3.2	83.5	53.0	7.0	**2.5**
Bolt	193.5	62.2	379.3	44.7	-	**4.6**

Table 1. Average center error (in pixels) with the best results shown in red fonts.

5. CONCLUSIONS

In this paper, we propose a novel fragment-based tracking framework using online multiple kernel learning (MKL) method. An online MKL for tracking is implemented by considering temporal continuity. The proposed tracker adaptively integrate the discriminative power of multiple fragments of the object. In addition, for better robustness we combine two kinds of independent features to complementarily represent the patches. Experiments on several challenging image sequences show that our proposed tracking framework achieves favorable performance.

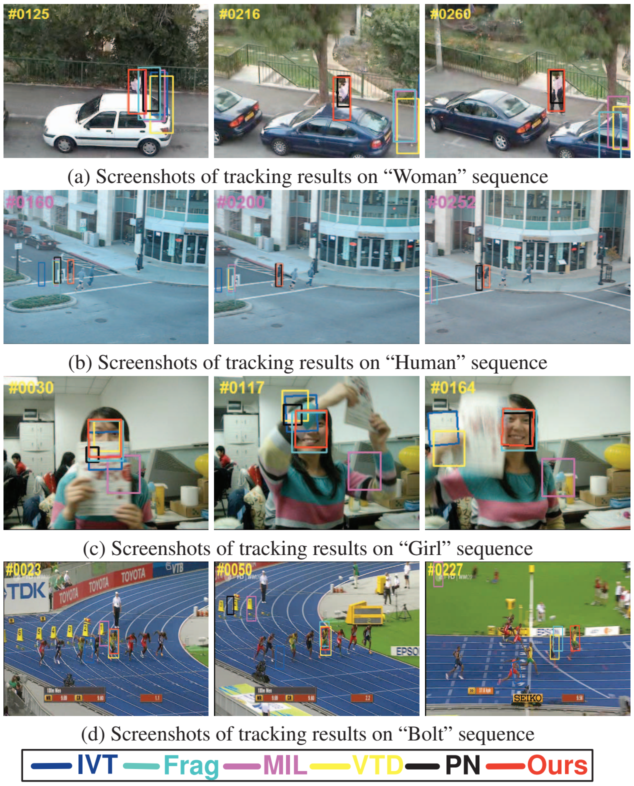
Acknowledgements This work is supported by the National Natural Science Foundation of China #61071209.

6. REFERENCES

[1] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *PAMI*, vol. 25, no. 5, pp. 564–575, 2003.

[2] R.-S. Lin, D. Ross, J. Lim, and M.-H. Yang, "Incremental learning for robust visual tracking," *IJCV*, vol. 77, pp. 125–141, 2008.

[3] F. Tang, S. Brennan, Q. Zhao, and H. Tao, "Co-tracking using semi-supervised support vector machines," in *ICCV*, 2007, pp. 1–8.



(a) Screenshots of tracking results on "Woman" sequence

(b) Screenshots of tracking results on "Human" sequence

(c) Screenshots of tracking results on "Girl" sequence

(d) Screenshots of tracking results on "Bolt" sequence

Fig. 4. Qualitative comparison with state-of-the-art methods.

[4] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *CVPR*, 2009, pp. 983–990.

[5] Z. Kalal, J. Matas, and K. Mikolajczyk, "P-N learning: Bootstrapping binary classifiers by structural constraints," in *CVPR*, 2010, pp. 49–56.

[6] S. Wang, H. Lu, F. Yang, and M.-H. Yang, "Superpixel tracking," in *ICCV*, 2011, pp. 1323–1330.

[7] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *CVPR*, 2006, pp. 798–805.

[8] F. Wang, S. Yu, and J. Yang, "A novel fragments-based tracking algorithm using mean shift," in *ICARCV*, 2008, pp. 694–698.

[9] C. Lampert and M. Blaschko, "A multiple kernel learning approach to joint multi-class object detection," in *DAGM-Symposium*, 2008, pp. 31–40.

[10] J. Yang, Y. Li, Y. Tian, L. Duan, and W. Gao, "Group-sensitive multiple kernel learning for object categorization," in *ICCV*, 2009, pp. 436–443.

[11] F. Yang, H. Lu, and Y.-W. Chen, "Human tracking by multiple kernel boosting with locality affinity constraints," in *ACCV*, 2010.

[12] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet, "Simplemkl," *Journal of Machine Learning Research*, vol. 9, pp. 2491–2521, 2008.

[13] A. Kembhavi, B. Siddiquie, R. Miezianko, S. McCloskey, and L. S. Davis, "Incremental multiple kernel. learning for object recognition," in *ICCV*, 2009, pp. 638–645.

[14] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005, pp. 886–893.

[15] D. Wang, H. Lu, and Y.-W. Chen, "Object tracking by multi-cues spatial pyramid matching," in *ICIP*, 2010, pp. 3957–3960.

[16] N. Cristianini, J. Shawe-Taylor, A. Elisseeff, and J. Kandola, "On kernel-target alignment," in *NIPS*, 2001, pp. 367–373.

[17] K. Grauman and T. Darrell, "The pyramid match kernel: Discriminative classification with sets of image features," in *ICCV*, 2005, pp. 1458–1465.

[18] J. Kwon and K. M. Lee, "Visual tracking decomposition," in *CVPR*, 2010, pp. 1269–1276.