CONE-RESTRICTED KERNEL SUBSPACE METHODS

Takumi Kobayashi, Fumito Yoshikawa and Nobuyuki Otsu

National Institute of Advanced Industrial Science and Technology
1-1-1 Umezono, Tsukuba, Japan

ABSTRACT

We propose cone-restricted kernel subspace methods for pattern classification. A cone is mathematically defined in a manner similar to a linear subspace with a nonnegativity constraint. Since the angles between vectors (i.e., inner products) are fundamental to the cone, kernel tricks can be directly applied. The proposed methods approximate the distribution of sample patterns by using the cone in kernel feature space via kernel tricks, and the classification is more accurate than that of the kernel subspace method. Due to the nonlinearity of kernel functions, even a single cone in the kernel feature space can can cope with multi-modal distributions in the original input space. In the experimental results on person detection and motion detection, the proposed methods exhibit the favorable performances.

Index Terms— Pattern classification, kernel-based method, cone, subspace method

1. INTRODUCTION

In recent years, linear methods for classifying pattern vectors have been extended to kernel-based methods, such as kernel principal component analysis [1] and kernel discriminant analysis [2], to deal with linearly inseparable pattern distributions. These kernel-based methods have performed well in various pattern recognition tasks in comparison with the linear methods. In kernel-based methods, input pattern vectors are implicitly embedded in high dimensional space, called kernel feature space, using kernel tricks [1], and then linear methods are applied in that space. Kernel subspace methods [3, 4], which assume a linear subspace in the kernel feature space, also provide favorable performances. The successful results indicate that pattern distribution in kernel feature space can be approximated by simple models, such as subspaces, even though the linearity in the original input space no longer holds due to the nonlinear kernel functions.

Cone-restricted subspace methods [5] were recently proposed and their superiority to the linear subspace method has been confirmed. The methods approximate a pattern distribution by using a cone. A cone is mathematically defined in a manner similar to a linear subspace with the constraint of nonnegativity that is compatible with the nonnegative features, such as histogram-based features [6, 7]. While the pattern vectors can be strictly classified at the boundary of the cone, cone-restricted subspace methods retain the favorable property of the subspace method, i.e., robustness to scale and additive changes of input pattern vectors. Therefore, the cone would be useful for approximating the pattern distribution even in kernel feature space.

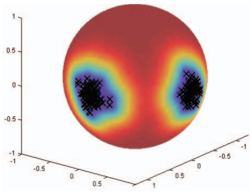


Fig. 1. A kernel cone for two modal distributions is shown in three-dimensional input space. The pseudo color indicates the angle to the kernel cone. This figure is best viewed in color.

In this paper, we extend the cone-restricted subspace methods to kernel-based methods by utilizing kernel tricks. The angles between pattern vectors (i.e., inner products) are fundamental to the cone, and thus kernel tricks are applicable. Thereby, a cone is also definable in kernel feature space. Due to the nonlinearity of kernel functions, the kernel cone can deal with the multi-modalities of pattern distributions in the original input space (Fig. 1), whereas the linear cone can only deal with uni-modal distributions. In this paper, we develop two types of kernel-based methods, using a strict convex cone and a circular cone [5] [1]. In experiments, we apply the proposed methods to one-class learning problems.

2. CONE-RESTRICTED KERNEL SUBSPACE METHODS

We propose two types of *cone-restricted kernel subspace methods* that utilize kernel tricks [1]. The proposed methods approximate the pattern distribution by using the cone in the kernel feature space. The first method is based on a convex cone defined in the kernel feature space and the second method is based on a circular (elliptic) cone.

[1] In [5], the method of covering convex cone is also proposed and it can be extended to kernel-based method. In this paper, however, we describe the kernel-based extensions of only a strict convex cone and a circular cone due to their superior performances to the covering cone method.

978-1-4244-7993-1/10/$26.00 ©2010 IEEE

2.1. Kernel convex cone

We first define the convex cone in kernel feature space, and then describe how to find the basis samples forming the kernel convex cone.

2.1.1. *Definition of kernel convex cone*

The kernel convex cone is defined as

$$C : \left\{ \boldsymbol{\phi}_x \ \mid \ \boldsymbol{\phi}_x = \sum_{i=1}^{N} \alpha_i \boldsymbol{\phi}(\boldsymbol{x}_i) = \boldsymbol{\Phi}(\boldsymbol{X})\boldsymbol{\alpha}, \ \ \alpha_i \geq 0 \right\}, \quad (1)$$

where $\boldsymbol{\phi}$ are nonlinear functions that map the input pattern vector $\boldsymbol{x}$ into higher dimensional space $\boldsymbol{\phi}(\boldsymbol{x})$ so that kernel function $k(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{\phi}(\boldsymbol{x})'\boldsymbol{\phi}(\boldsymbol{y})$, $\boldsymbol{\Phi}(\boldsymbol{X}) = [\boldsymbol{\phi}(\boldsymbol{x}_1), .., \boldsymbol{\phi}(\boldsymbol{x}_N)]$, N is the number of sample vectors $\boldsymbol{x}_i$, and α_i are nonnegative coefficients, $\boldsymbol{\alpha} = [\alpha_1, , \alpha_N]' \in \boldsymbol{R}^{N\times 1}$. Cone C is embedded in higher dimensional space, called the kernel feature space, via the functions $\boldsymbol{\phi}$. In the proposed method, an input vector $\boldsymbol{y}$ is classified based on the angle θ between the transformed vector $\boldsymbol{\phi}(\boldsymbol{y})$ and the vector perpendicularly projected onto the kernel convex cone C in the kernel feature space. The angle θ is calculated as follows:

$$\begin{aligned} \theta &= \arcsin \frac{\min_{\boldsymbol{\phi}_x \in C} ||\boldsymbol{\phi}(\boldsymbol{y}) - \boldsymbol{\phi}_x||}{||\boldsymbol{\phi}(\boldsymbol{y})||} \\ &= \arcsin \frac{\sqrt{\min_{\boldsymbol{\alpha} \geq 0} ||\boldsymbol{\phi}(\boldsymbol{y}) - \boldsymbol{\Phi}(\boldsymbol{X})\boldsymbol{\alpha}||^2}}{||\boldsymbol{\phi}(\boldsymbol{y})||} \\ &= \arcsin \sqrt{\frac{\min_{\boldsymbol{\alpha} \geq 0} \boldsymbol{\alpha}'\boldsymbol{K}\boldsymbol{\alpha} - 2\boldsymbol{\alpha}'\boldsymbol{k_y} + k(\boldsymbol{y}, \boldsymbol{y})}{k(\boldsymbol{y}, \boldsymbol{y})}}, \end{aligned} \quad (2)$$

where $0 \leq \theta \leq \pi/2$, $\boldsymbol{K} \in \boldsymbol{R}^{N\times N}$ is a kernel Gram matrix ($K_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$), $\boldsymbol{k_y} \in \boldsymbol{R}^{N\times 1}$ is a vector consisting of $\{\boldsymbol{k_y}\}_i = k(\boldsymbol{x}_i, \boldsymbol{y})$. This optimization problem is efficiently solved by applying the nonnegative least square (NNLS) method [8] to matrix $\boldsymbol{K}$ and vector $\boldsymbol{k_y}$. However, it is time-consuming to use all of the samples $\boldsymbol{\Phi}(\boldsymbol{X})$ as bases of the kernel cone, for calculating the angle in Eq.(2). To reduce the computational cost, we extract a smaller number of basis samples by eliminating the redundant ones.

2.1.2. *Basis samples in kernel convex cone*

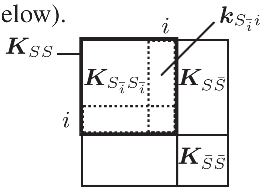
First, we define convex-redundant samples $\boldsymbol{\phi}(\boldsymbol{x}_i)$ by

$$\boldsymbol{\phi}(\boldsymbol{x}_i) = \sum_{j \neq i} \boldsymbol{\phi}(\boldsymbol{x}_j)\boldsymbol{\alpha}_j = \boldsymbol{\Phi}(\boldsymbol{X}_{\bar{i}})\boldsymbol{\alpha}_{\bar{i}}, \ \ s.t. \ \boldsymbol{\alpha}_{\bar{i}} \geq 0, \quad (3)$$

where $\bar{i}$ denotes the index set excluding i. Even if the redundant samples are excluded, the kernel convex cone can be represented by the remaining subset of samples without collapsing the cone. The redundancy is evaluated based on the angle between the vector $\boldsymbol{\phi}(\boldsymbol{x}_i)$ and the cone composed of $\boldsymbol{\Phi}(\boldsymbol{X}_{\bar{i}})$; a smaller angle indicates higher redundancy of the sample (Eq.(3) holds for a zero angle). Thus, based on the kernel Gram matrix $\boldsymbol{K}$, the redundant samples are sequentially eliminated in a manner of leave-one-out using Eq.(2), as in reducing support vectors [9]. The algorithm is described in Algorithm 1. The basis samples, which are similar to the support vectors in support vector machines (SVM) [10], are obtained in this greedy manner, and so the computational cost in Eq.(2) is drastically reduced.

Algorithm 1 : Extraction of basis samples in convex cone

```
1: Select subset $S$ randomly from the whole index set $W = \{1, \cdots, N\}$ (complementary index set is denoted by $\bar{S}$).
2: Calculate the angle $\theta_i$ between each sample $\boldsymbol{\phi}(\boldsymbol{x}_i)$ ($i \in S$) and the cone $C_{S_{\bar{i}}}$ by using kernel matrix $\boldsymbol{K}_{S_{\bar{i}}S_{\bar{i}}}$ and vector $\boldsymbol{k}_{S_{\bar{i}}i}$ in Eq.(2) (see below).
```


$\boldsymbol{K}_{SS}$
i
$\boldsymbol{k}_{S_{\bar{i}}i}$
$\boldsymbol{K}_{S_{\bar{i}}S_{\bar{i}}}$
$\boldsymbol{K}_{S\bar{S}}$
i
$\boldsymbol{K}_{\bar{S}\bar{S}}$

```
3: Eliminate the index $\{i|\theta_i < \theta_{\text{thre}}(\approx 0)\}$ from $W$ and $S$ (the sample is *convex-redundant*).
4: Calculate the angle $\theta_j$ between sample $\boldsymbol{\phi}(\boldsymbol{x}_j)$ ($j \in \bar{S}$) and cone $C_S$ by using $\boldsymbol{K}_{SS}$ and $\boldsymbol{k}_{Sj}$, and eliminate index $\{j|\theta_j < \theta_{\text{thre}}\}$ from $W$.
5: Repeat steps 1∼4 until no redundant sample index is found.
```

2.2. Kernel circular cone

In the second proposed method, the sample distribution is approximated by a circular (elliptic) cone in the kernel feature space. Suppose that all samples are projected onto a unit (kernel) hypersphere. Then the kernel circular cone is simply defined by

$$\left\{ \boldsymbol{x} | \frac{\boldsymbol{\phi}_\mu' \boldsymbol{\phi}(\boldsymbol{x})}{\sqrt{\boldsymbol{\phi}(\boldsymbol{x})'\boldsymbol{\phi}(\boldsymbol{x})}} \geq b \right\}, \quad (4)$$

where $\boldsymbol{\phi}_\mu$ is the orientation vector of the kernel circular cone, and $\boldsymbol{\phi}_\mu'\boldsymbol{\phi}(\boldsymbol{x}) = b$ is the hyperplane perpendicular to $\boldsymbol{\phi}_\mu$ that intersects the unit hypersphere. The primary problem for the kernel circular cone is determining the hyperplane.

2.2.1. *Whitening*

Since the samples are generally distributed elliptically, they should be whitened in the kernel feature space before applying the kernel circular cone. The whitening is performed by applying kernel principal component analysis (KPCA) to the samples projected onto the unit hypersphere. In this case, the orientation vector is simply assumed to be the sample mean vector $\boldsymbol{\phi}_m$:

$$\boldsymbol{\phi}_m = \frac{\tilde{\boldsymbol{\Phi}}(\boldsymbol{X})\mathbf{1}}{\sqrt{\mathbf{1}'\tilde{\boldsymbol{\Phi}}(\boldsymbol{X})'\tilde{\boldsymbol{\Phi}}(\boldsymbol{X})\mathbf{1}}} = \boldsymbol{\Phi}(\boldsymbol{X}) \underbrace{\text{diag}(\boldsymbol{K})^{-1/2} \frac{\mathbf{1}}{\sqrt{\mathbf{1}'\tilde{\boldsymbol{K}}\mathbf{1}}}}_{\boldsymbol{m} \in \boldsymbol{R}^{N\times 1}},$$

where $\tilde{\boldsymbol{\phi}}(\boldsymbol{x}) = \frac{\boldsymbol{\phi}(\boldsymbol{x})}{||\boldsymbol{\phi}(\boldsymbol{x})||}$, and $\tilde{\boldsymbol{K}} = \text{diag}(\boldsymbol{K})^{-\frac{1}{2}}\boldsymbol{K}\text{diag}(\boldsymbol{K})^{-\frac{1}{2}}$ is a normalized Gram matrix. The samples are assumed to be elliptically distributed around this orientation vector and we project them into the complementary subspace to $\boldsymbol{\phi}_m$ by

$$\hat{\boldsymbol{\phi}}(\boldsymbol{x}_i) = (\boldsymbol{I} - \boldsymbol{\phi}_m\boldsymbol{\phi}_m')\frac{\boldsymbol{\phi}(\boldsymbol{x}_i)}{||\boldsymbol{\phi}(\boldsymbol{x}_i)||} = (\boldsymbol{I} - \boldsymbol{\phi}_m\boldsymbol{\phi}_m')\tilde{\boldsymbol{\phi}}(\boldsymbol{x}_i).$$

Then, KPCA is applied to these projected samples:

$$\hat{\boldsymbol{K}}\hat{\boldsymbol{K}}\boldsymbol{\alpha} = \lambda\hat{\boldsymbol{K}}\boldsymbol{\alpha} \quad \therefore \hat{\boldsymbol{K}}\boldsymbol{\alpha} = \lambda\boldsymbol{\alpha}, \quad s.t. \;\; \boldsymbol{\alpha}'\hat{\boldsymbol{K}}\boldsymbol{\alpha} = 1, \quad (5)$$

where $\hat{\boldsymbol{K}} = \tilde{\boldsymbol{K}} - \frac{\tilde{\boldsymbol{K}}\boldsymbol{1}\boldsymbol{1}'\tilde{\boldsymbol{K}}'}{\boldsymbol{1}'\tilde{\boldsymbol{K}}\boldsymbol{1}}$. The j-th principal axis is

$$\hat{\boldsymbol{\Phi}}(\boldsymbol{X})\boldsymbol{\alpha}_j = \boldsymbol{\Phi}(\boldsymbol{X})\underbrace{\text{diag}(\boldsymbol{K})^{-1/2}\left[\boldsymbol{\alpha}_j - \frac{\boldsymbol{1}'\tilde{\boldsymbol{K}}\boldsymbol{\alpha}_j}{\boldsymbol{1}'\tilde{\boldsymbol{K}}\boldsymbol{1}}\boldsymbol{1}\right]}_{\boldsymbol{a}_j \in \boldsymbol{R}^{N\times 1}}. \quad (6)$$

Thus, by using n principal axes, the mapping vectors for whitening are obtained as

$$\boldsymbol{V} = [\boldsymbol{\phi}_m, \hat{\boldsymbol{\Phi}}(\boldsymbol{X})\boldsymbol{\alpha}_1, \cdots, \hat{\boldsymbol{\Phi}}(\boldsymbol{X})\boldsymbol{\alpha}_n] = \boldsymbol{\Phi}(\boldsymbol{X})[\boldsymbol{m}, \boldsymbol{A}], \quad (7)$$

where $\boldsymbol{A} = [\boldsymbol{a}_1, .., \boldsymbol{a}_n] \in \boldsymbol{R}^{N\times n}$. The samples are whitened by

$$\check{\boldsymbol{x}}_i = \underbrace{\text{diag}\left(1, 1, \sqrt{\frac{\lambda_1}{\lambda_2}}, \cdots, \sqrt{\frac{\lambda_1}{\lambda_n}}\right)}_{\boldsymbol{S} \in \boldsymbol{R}^{(n+1)\times(n+1)}}\boldsymbol{V}'\boldsymbol{\phi}(\boldsymbol{x}_i) = \boldsymbol{S}[\boldsymbol{m}, \boldsymbol{A}]'\boldsymbol{k}_i.$$

In this whitening, the sample distribution is scaled based on the standard deviation along the first principal axis. Vector $\check{\boldsymbol{x}}$ is further normalized, as described in the next section.

2.2.2. *Determination of hyperplane*

After whitening by KPCA, the sample vectors are embedded in Euclidean space $\check{\boldsymbol{x}}$. As in the method of a linear circular cone [5], the optimal plane for the kernel circular cone is obtained by applying (linear) 1-class SVM to the whitened samples:

$$\min_{\boldsymbol{\mu},b} \frac{1}{2}||\boldsymbol{\mu}||^2 - b, \qquad s.t. \;\; \boldsymbol{\mu}'\frac{\check{\boldsymbol{x}}_i}{||\check{\boldsymbol{x}}_i||} \geq b. \quad (8)$$

The solutions are then normalized: $\boldsymbol{\mu} \leftarrow \frac{\boldsymbol{\mu}}{||\boldsymbol{\mu}||}, b \leftarrow \frac{b}{||\boldsymbol{\mu}||}$.

2.2.3. *Angle to kernel circular cone*

The angle to the kernel circular cone is easily calculated due to its simple formulation in Eq.(4). We calculate the projected vector $\boldsymbol{\zeta}$ of an input vector $\boldsymbol{y}$ onto the kernel circular cone. Since the kernel circular cone is defined in the kernel subspace by whitening (KPCA), input vector $\boldsymbol{\phi}(\boldsymbol{y})$ is first projected into the kernel subspace by using $\boldsymbol{V}$:

$$\check{\boldsymbol{y}} = \boldsymbol{S}\boldsymbol{V}'\boldsymbol{\phi}(\boldsymbol{y}) = \boldsymbol{S}[\boldsymbol{m}, \boldsymbol{A}]'\boldsymbol{k}_{\boldsymbol{y}}. \quad (9)$$

Then, $\boldsymbol{\zeta}$ is obtained as follows:

$$\begin{aligned} \check{\boldsymbol{\zeta}} &= \cos\theta_C\boldsymbol{\mu} + \sin\theta_C\frac{\check{\boldsymbol{y}} - (\check{\boldsymbol{y}}'\boldsymbol{\mu})\boldsymbol{\mu}}{||\check{\boldsymbol{y}} - (\check{\boldsymbol{y}}'\boldsymbol{\mu})\boldsymbol{\mu}||} \quad (10) \\ \boldsymbol{\zeta} &= \boldsymbol{V}\boldsymbol{S}^{-1}\check{\boldsymbol{\zeta}}, \quad (11) \end{aligned}$$

where $\theta_C (= \arccos(b))$ is the spread angle of the kernel circular cone, and $\check{\boldsymbol{\zeta}}$ is the vector projected from $\boldsymbol{\zeta}$ into the kernel subspace. Thus, we can compute the angle between the input vector and kernel circular cone as follows:

$$\theta = \arccos\frac{\boldsymbol{\phi}(\boldsymbol{y})}{||\boldsymbol{\phi}(\boldsymbol{y})||}\frac{\boldsymbol{\zeta}}{||\boldsymbol{\zeta}||} = \arccos\frac{\check{\boldsymbol{y}}'\boldsymbol{S}^{-2}\check{\boldsymbol{\zeta}}}{\sqrt{k(\boldsymbol{y}, \boldsymbol{y})}\sqrt{\check{\boldsymbol{\zeta}}'\boldsymbol{S}^{-2}\check{\boldsymbol{\zeta}}}}.$$

Since the dimensionality of the kernel subspace is low, the computational cost for calculating the angle is also low, even if the kernel feature space has high dimensionality.

3. EXPERIMENTAL RESULT

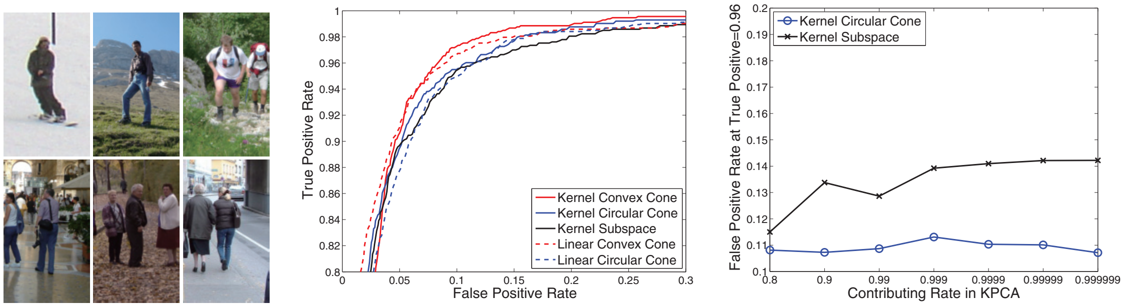
We applied the proposed methods to two kinds of one-class learning problems: person detection and motion detection. In one-class learning, only the positive samples to be detected are used to construct the kernel cone, and then the input sample is classified based on the angle to the cone.

3.1. Person detection

We used the INRIA person dataset [11] containing person images (64x128) with large variability, as shown in Fig. 2(a). We selected 2416 person images for training, and 1132 person images and 25770 person-free images for testing. The gradient local auto-correlation (GLAC) image features [6] were extracted from these images and the RBF kernel was applied to the features. The cone-restricted kernel subspace methods were compared to the methods of the linear cone [5] and the kernel subspace [3]. The results are shown in Fig. 2(b). The cone-restricted kernel subspace methods show an improvement over the performances of the linear cones and the kernel subspace method. As shown in Fig. 2(c), the kernel circular cone method is robust with respect to the dimensionality of the kernel subspace for whitening, in contrast to the kernel subspace method. These results show that the cone-based method is effective even in the kernel feature space.

3.2. Motion detection

Next, we used sports motion images for motion detection. The task is to detect weightlifting motions from a long motion image sequence captured by a stationary camera at a practice field. The test image sequence has approximately 30000 frames containing 14 weightlifting motions performed by several players as well as various irrelevant motions, as shown in Fig. 3(a). For training, we collected 38 short motion images containing only the target (weightlifting) motions captured at the different practice field from that of the test sequence. We employed the kernel circular cone method, since the method is favorable in terms of performance and computational cost, and applied the RBF kernel to cubic higher-order local auto-correlation (CHLAC) motion features [7] extracted from the motion images. To detect the target motions, we applied the detection window with 75, 150 and 300 frames along the time



(a) Images (b) ROC (c) Dimensionality

Fig. 2. Results of person detection.

axis, and performed non-maximum suppression [11] as post-processing. The detection performance was evaluated by the average precision rate. The performance of the kernel circular cone was 99.5%, whereas the performance of the linear circular cone was 98.5%. The detection results of the kernel circular cone method are shown in Fig. 3(b, c). As shown in Fig. 3(b), we can easily determine the threshold for detecting all the weightlifting motions except for the last one, in which the player failed to lift the barbell. Although the eighth motion is redundantly detected, the detected motion periods are quite close to the ground truth, as shown in Fig. 3(c).

4. CONCLUSION

We proposed two cone-restricted kernel subspace methods. The methods approximate the sample distributions in kernel feature space by a cone and strictly classify the patterns at the boundary of the cone, compared to kernel subspace methods. In addition, multi-modalities in the original input space can be dealt with in virtue of nonlinear kernel functions. The experimental results for person detection and motion detection demonstrated the effectiveness of the proposed methods.

5. REFERENCES

[1] B. Schölkopf and A. Smola, Eds., *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, 2002.

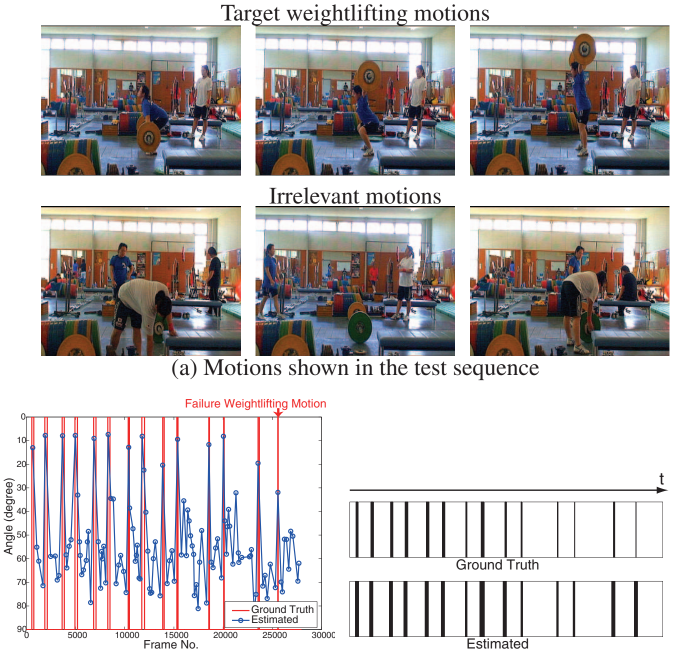
[2] S. Mika, G. Ratsch, J. Weston, B. Schölkopf, and K. Müller, "Fisher discriminant analysis with kernels," in *IEEE Neural Networks for Signal Processing Workshop*, 1999, pp. 41–48.

[3] E. Maeda and H. Murase, "Kernel-based nonlinear subspace method for pttern recognition," *Systems and Computers in Japan*, vol. 33, no. 1, 2002.

[4] K. Fukui and O. Yamaguchi, "The kernel orthogonal mutual subspace method and its application to 3d object recognition," in *ACCV*, 2007.

[5] T. Kobayashi and N. Otsu, "Cone-restricted subspace method," in *ICPR*, 2008.

[6] T. Kobayashi and N. Otsu, "Image feature extraction using gradient local auto-correlations," in *ECCV*, 2008.



(a) Motions shown in the test sequence

(b) Angle to the cone (c) Detection result

Fig. 3. Results of weightlifting motion detection. (a) examples of target and irrelevant motions, (b) detection results: the blue points are the center time points of the estimated motion periods and the red line indicates the ground truth periods, and (c) detected periods: the black bars indicate periods of weightlifting motion.

[7] T. Kobayashi and N. Otsu, "A three-way auto-correlation based approach to motion recognition," *Pattern Recognition Letters*, vol. 30, no. 3, pp. 185–192, 2009.

[8] R. Bro and S.D Jong, "A fast non-negativity-constrained least squares algorithm," *Journal of Chemometrics*, vol. 11, 1997.

[9] T. Kobayashi and N. Otsu, "Efficient reduction of support vectors in kernel-based methods," in *ICIP*, 2009.

[10] V.N. Vapnik, *Statistical Learning Theory*, Wiley, 1998.

[11] Navneet Dalal and Bill Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005.