View-Invariant Action Recognition using Latent Kernelized Structural SVM

Xinxiao Wu and Yunde Jia

Beijing Laboratory of Intelligent Information Technology
School of Computer Science, Beijing Institute of Technology
Beijing 100081, P.R. China
{wuxinxiao, jiayunde}@bit.edu.cn

Abstract. This paper goes beyond recognizing human actions from a fixed view and focuses on action recognition from an arbitrary view. A novel learning algorithm, called latent kernelized structural SVM, is proposed for the view-invariant action recognition, which extends the kernelized structural SVM framework to include latent variables. Due to the changing and frequently unknown positions of the camera, we regard the view label of action as a latent variable and implicitly infer it during both learning and inference. Motivated by the geometric correlation between different views and semantic correlation between different action classes, we additionally propose a mid-level correlation feature which describes an action video by a set of decision values from the pre-learned classifiers of all the action classes from all the views. Each decision value captures both geometric and semantic correlations between the action video and the corresponding action class from the corresponding view. After that, we combine the low-level visual cue, mid-level correlation description, and high-level label information into a novel nonlinear kernel under the latent kernelized structural SVM framework. Extensive experiments on multi-view IXMAS and MuHAVi action datasets demonstrate that our method generally achieves higher recognition accuracy than other state-of-the-art methods.

Key words: View-invariant action recognition, latent kernelized structural SVM, correlation feature, multiple level features

1 Introduction

Automatic recognition of human action from a single video has become an essential area of research in computer vision. Many previous approaches [1] [2] [3] [4] [5] have achieved good performance, however, the assumption that all the action videos are captured from a fixed view point limits their robustness to different view points and camera parameters in real world applications. Consequently, view-invariant action recognition from an arbitrary view point has attracted much attention in recent years. Because of the changing positions of cameras and self-occlusions between different body parts, the appearance and motion of actions may drastically vary from one view point to another. Therefore, view-invariant action recognition poses substantial challenges for computer vision algorithms.

Some existing approaches address the view-invariant action recognition by using epipolar geometry [6] [7] or a full 3D reconstruction [8] [9]. Such approaches require either the point correspondence estimation or the calibration setup of multiple cameras. Some other methods propose to learn the view-invariant features such as the temporal self-similarity descriptor [10] and the view and style-independent manifold representation [11]. However, both of them rely on the rough localization and tracking of people in the video.

In this work, we address the view-invariant action recognition from a different perspective by avoiding many assumptions of previous methods. We propose a novel latent kernelized structural SVM learning algorithm that allows the use of latent variables in the kernelized structural SVM for recognizing actions from an arbitrary view point. This method discriminatively learns the mapping function from a single video to an action class. In order to address the difficulty of frequently changing and unknown positions of camera, we treat the view label of an action as a latent variable and implicitly infer it during both learning and inference stages. Consequently, the view-invariant recognition in this paper is achieved by unifying action classification and view prediction in a principled structural framework.

Due to the geometric constraints between different views, videos of the same action recorded from multiple views may demonstrate some correlations. For example, the videos from neighboring cameras may present more similar on visual cues when compared with the videos from faraway cameras, so it is beneficial to describe actions by exploiting their correlations between different views. Moreover, we also introduce the semantic correlations between different action classes to represent actions. Specifically, a set of decision values produced by the pre-learned classifiers of all the action classes from all the view points is proposed as a mid-level correlation feature in this paper. Each decision value measures the likelihood that the action video belongs to the corresponding action class from the corresponding view.

Moreover, we define a novel nonlinear kernel to fuse multiple level information under the latent structural SVM framework for further improving the recognition performance. This kernel combines multiple kernels and measures the similarity between two action videos based on the low-level visual feature, the mid-level correlation feature and the high-level action class-view label pair information.

The main contributions of this work are three-fold. Firstly, we propose a new latent kernelized structural SVM learning method for view-invariant action recognition by regarding the view label of action as a latent variable. Secondly, we propose a novel mid-level correlation feature with more discriminative power and robustness, which captures both geometric correlation between multiple views and semantic correlation between different action classes. Thirdly, we integrate the low-level visual cue, mid-level correlation feature and high-level class-view label pair information into a novel nonlinear kernel used in the latent kernelized structural SVM framework.

2 Related Work

2.1 View-invariant Action Recognition

Yilmaz and Shah [6] exploited dynamic epipolar geometry by imposing temporal fundamental matrix for view-invariant action recognition. Shen and Foroosh [7] proposed the ratios among the elements in the upper left 2×2 submatrix of fundamental matrix F for action recognition from varying viewpoints. Such epipolar geometry-based methods assume that the point correspondence should be known, which is still a difficult problem.

In [8], the actions are described by 3D exemplars represented by visual hulls, and action recognition is achieved by matching between observation and exemplars in 2D by projecting visual hulls. Yan et al. [9] proposed 4D action feature model to recognize actions in arbitrary views by mapping features from individual views to the surfaces of 4D action shapes obtained from time ordered multi-view 3D reconstructions of the actors. These 3D construction-based approaches require a calibration setup of multiple cameras, which restricts their applicability in practice.

Junejo et al. [10] explored self-similarities of action sequences over time and extracted view-invariant features based on frame-to-frame similarities within a sequence. In [11], each action is modeled as the embedded manifold of image sequences by dimension reduction methods, and all view-dependent manifolds are automatically combined to discover a unified and view-independent representation. With the assumption of rough localization and tracking of people in the video, these methods are usually applied in constrained environments.

Recently, transfer learning has been exploited to recognize actions from the target view when training the action models from the source view. Liu et al. [12] proposed to learn bilingual-words from two view-dependent vocabularies and transferred actions from bag-of-visual-words model to bag-of-bilingual-words model. Farhadi and Tabrizi [13] used Maximum Margin Clustering to generate split features in the source view and then learned the split features in the target view by the transferred split values from source view.

2.2 Discriminative Structural Learning Model

The discriminative structural learning methods most closely related to our approach are that of [14] [15] [16]. Tsochantaridis et al. [14] generalized the multiclass SVM learning to the broader problem of learning the complex structured outputs and formulated the objective function as a dual formulation allowing the use of kernel functions. Yu and Joachims [16] used approximate cutting planes and random sampling to enable efficient training of structured SVM with kernels. Different from [14] and [16], we extend the kernelized structural SVM to include latent variables for structured output prediction.

Yu and Joachims [15] presented the structural SVM with latent variables where the feature vector extracted jointly from inputs and outputs is akin to conventional linear SVMs, and utilized the Concave-Convex Procedure to solve the optimization problem. Latent structural SVM and its variants have found their wide applications in many computer vision scenarios, e.g. object recognition with attributes [17], image annotation and segmentation [18], action recognition from pose estimation [19], and group activity recognition [20]. All these methods utilize linear models and adopt a non-convex cutting plane algorithm [21] to solve the optimization problem. In contrast, our method bases the optimization on the dual program formulation and uses the nonlinear kernel function to fuse multiple level information with more flexible and powerful input-output representations.

3 Latent Kernelized Structural SVM for View-Invariant Action Recognition

3.1 Model Formulation

We define the view-invariant action recognition as learning a prediction function that maps the input action video to an output action label with the unobserved latent view label. Suppose we are given a training set of structured pairs $\{(x_i, y_i)\}, i = 1, 2, ..., n, (x_i, y_i) \in \mathbf{X} \times \mathbf{Y}$ and a set of unobserved latent variables $\{h_i\}, i = 1, 2, ..., n, h_i \in \mathbf{H}$, our goal is to learn a prediction rule of the following form:

$$f_{\mathbf{w}}(x) = \arg\max_{(y,h)\in\mathbf{Y}\times\mathbf{H}} F(x, y, h) = \arg\max_{(y,h)\in\mathbf{Y}\times\mathbf{H}} [\mathbf{w} \cdot \Phi(x, y, h)], \quad (1)$$

where $\Phi(x, y, h)$ is a joint feature vector that describes the relationship among the input action video x, output action class label y and latent view label h. The optimization problem of computing this $\arg\max$ is typically referred to as the "inference" or "recognition" problem. We define $F(x, y, h)$ as follows:

$$F(x, y, h) = \mathbf{w} \cdot \Phi(x, y, h) = \eta \cdot \phi(x, y) + \beta \cdot \varphi(x, y, h). \quad (2)$$

The model parameters $\mathbf{w}$ are simply the concatenation of two parts, i.e., $\mathbf{w} = \{\eta; \beta\}$. The details of each term are described in the following.

Global view-invariant action model $\eta \cdot \phi(x, y)$: This potential function measures the compatibility between an action video x and an action class label y without considering the view point information. It is parameterized as: $\eta \cdot \phi(x, y) = \sum_{i=1}^{|\mathbf{Y}|} \eta_i \cdot \delta(x) \cdot \mathrm{I}_i(y)$, where $\delta(x)$ represents the extracted feature of action x, and η_i indicates the weight vector for the feature $\delta(x)$ to take the action label i. $\mathrm{I}_i(y)$ is an indicator function, namely, $\mathrm{I}_i(y) = 1$ if $y = i$, and $\mathrm{I}_i(y) = 0$ otherwise.

Local view-specific action model $\beta \cdot \varphi(x, y, h)$: In addition to the global view-invariant action model, we also define a view-specific action model parameterized by $\beta \cdot \varphi(x, y, h) = \sum_{i=1}^{|\mathbf{Y}|} \sum_{j=1}^{|\mathbf{H}|} \beta_{ij} \cdot \delta(x) \cdot \mathrm{I}_i(y) \cdot \mathrm{I}_j(h)$, where β_{ij} represents the weight vector for $\delta(x)$ to take the action label i when the view label is j. The motivation for this potential function is that the same action might appear differently across multiple views. By separately learning action models for each view, the learning becomes easier since the positive examples within the same action class are similar to each other.

3.2 Learning

Given a set of training examples $\{(x_i, y_i)\}, i = 1, 2, ..., n$, the model parameters $\mathbf{w}$ are learned through the following optimization problem:

$$\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{i=1}^{n} \xi_i,$$

$$\xi_i = l(\max_{\bar{h}_i \in \mathbf{H}} F(x_i, y_i, \bar{h}_i) - \max_{(\hat{y}_i, \hat{h}_i) \in \mathbf{Y} \times \mathbf{H}} F(x_i, \hat{y}_i, \hat{h}_i)), \tag{3}$$

where $l(t)$ is the hinge loss function defined by $l(t) = C \max(0, 1 - t)$. Since the objective function in Eq.(3) is a non-convex problem, we propose an algorithm that alternates between computing the latent variables $\bar{h}_i$ that best explains the training pair (x_i, y_i) and solving the standard structural SVM optimization problem while treating the latent variables as completely observed. By replacing the label pair (y, h) with s, the objective function of structural SVM with the observed latent variable $\bar{h}_i$ can be rewritten as

$$\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{i=1}^{n} \xi_i, \quad \xi_i = l(F(x_i, s_i) - \max_{\hat{s}_i \in \mathbf{Y} \times \mathbf{H}} F(x_i, \hat{s}_i)), \tag{4}$$

where $s_i = (y_i, \bar{h}_i)$ and $\hat{s}_i = (\hat{y}_i, \hat{h}_i)$. Different from [15], we base the optimization of structural SVM on the dual program formulation which only depends on inner products in the joint feature space allowing the use of kernel functions. Following [22], the dual optimization problem of standard structural SVM is formulated by

$$\min_{\alpha, \gamma} \gamma - \sum_i \alpha_{is_i},$$

$$\text{s.t. } \forall i : 0 \leq \alpha_{is_i} \leq C, \quad \forall i : \forall u \neq s_i : \alpha_{iu} \leq 0, \quad \forall i : \sum_{u \in \mathbf{S}} \alpha_{iu} = 0,$$

$$\forall u : \sum_i \alpha_{iu} = 0, \quad \gamma \geq \frac{1}{2} \sum_{i,j} \sum_{u,v \in \mathbf{S}} \alpha_{iu} \alpha_{jv} K(\psi(x_i, u), \psi(x_j, v)), \tag{5}$$

where $\mathbf{S} = \mathbf{Y} \times \mathbf{H}$. The solution of this dual problem gives a set of weights α for the support vectors. The kernel $K(\psi(x_i, u), \psi(x_j, v))$ analytically describes the relationship between two video-label pairs without requiring an explicit expression for the joint feature vector $\psi(x, s)$. After obtaining the optimal α, the scoring function can be given by

$$F(x, y, h) = F(x, s) = \sum_i \sum_{u \in \mathbf{S}} \alpha_{iu} K(\psi(x_i, u), \psi(x, s)). \tag{6}$$

The algorithm of latent kernelized structural SVM is listed in Algorithm 1.

Algorithm 1 Latent Kernelized Structural SVM

```
Input: $\{x_i, y_i\}_{i=1}^n$
Output: $\{\alpha_{iu}\}_{i=1}^n, \quad u \in \mathbf{S}$
Initialize $\{h_i^{(0)}\}_{i=1}^n$ and set $t = 0$.
repeat
    Compute the weights $\alpha_{iu}^{(t)}$ by solving Eq.(5) given the training set $\{x_i, y_i, h_i^{(t)}\}$.
    Compute the latent variables $h_i^{(t+1)}$ by solving $h_i^{(t+1)} = \arg\max_{h_i^{(t+1)}} \sum_j \sum_{u \in \mathbf{S}} \alpha_{ju}^{(t)} K(\psi(x_j, u_j^{(t)}), \psi(x_i, u_i^{(t+1)}))$ given the weights $\alpha_{iu}^{(t)}$ with $u_j^{(t)} = (y_j, h_j^{(t)})$ and $u_i^{(t+1)} = (y_i, h_i^{(t+1)})$.
until $\sum_i \sum_{u \in \mathbf{S}} |\alpha_{iu}^{(t+1)} - \alpha_{iu}^{(t)}| < \epsilon$
```

3.3 Inference

The inference problem is to find the best action label y for a test video x, and we need to solve the following optimization problem:

$$\max_{y,h} F(x, y, h) = \max_{s} F(x, s) = \max_{s \in \mathbf{S}} \sum_i \sum_{u \in \mathbf{S}} \alpha_{iu} K(\psi(x_i, u), \psi(x, s)). \quad (7)$$

For simplicity, we directly enumerate all the possible action class-view label pairs (y, h) to predict the optimal action label y for x.

4 Designing a Kernel by Fusing Multiple Level Information

Under the latent kernelized structural SVM framework, the joint kernel K is equivalent to the tensor product of the feature spaces produced by each individual kernel:

$$K(\psi(x_1, u_1), \psi(x_2, u_2)) = K_x(x_1, x_2) * K_u(u_1, u_2), \quad (8)$$

where $K_x(x_1, x_2)$ measures the video/action similarity and $K_u(u_1, u_2)$ measures the action class-view label pair similarity. Such joint kernel function fuses bottom-up video cues (i.e.,K_x) and top-down semantic label information (i.e.,K_u). It encodes the mutual matching between two video-class-view triples: if the videos are similar, the class-view label pairs have to be similar as well. In case either the videos are significantly different or the class-view label pairs are not matching, the kernel response has to be low. In this paper, we propose a novel mid-level correlation feature to describe actions and thus the video similarity kernel is designed to be the combination of low-level visual feature kernel $K_x^{low}(x_1, x_2)$ and mid-level correlation feature kernel $K_x^{mid}(x_1, x_2)$, i.e., $K_x(x_1, x_2) = K_x^{low}(x_1, x_2) + K_x^{mid}(x_1, x_2)$.

4.1 Low-level Visual Feature Kernel

The low-level visual feature kernel combines several individual kernels with each one capturing a specific type of visual feature:

$$K_x^{low}(x_1, x_2) = \sum_{l=1}^{L} k_l^{low}(x_1, x_2), \tag{9}$$

where L is the number of feature types and $k_l^{low}(x_1, x_2)$ is the kernel based on the l-th visual feature. To capture the motion and appearance information of actions, we respectively extract the spatio-temporal context distribution feature and appearance feature of interest points [23]. The dense trajectory features (including trajectory, HOG, HOF and MBH) proposed by Wang et al. [24] are also integrated to further improve the recognition performance. Moreover, the local SIFT feature [25] is extracted from randomly selected frames in the video and employed as a static description for action. Consequently, we use seven different types of heterogeneous and complementary low-level visual features (i.e., spatio-temporal context distribution and appearance features of interest points, four types of dense trajectory features, and SIFT feature) and the corresponding seven individual kernels are combined into the low-level visual feature kernel (i.e., $L = 7$).

4.2 Mid-level Correlation Feature Kernel

The extracted low-level visual features only represent the visual information of action video and their discriminative capability is limited. Thus we propose a mid-level correlation feature which captures the correlations between different action classes from different views, to abstract the visual content of video. Different from previous mid-level semantic feature such as concept score [26] and attribute feature [27] [28], the proposed correlation feature not only describes the semantic correlation between different classes, but also represents the geometric relationship between different views. The intuitive explanation is that: the same action captured by multiple views may often have deformation correlations due to the geometric constraints between views. For example, the same action may look similar with less deformation when observed by two neighboring views (e.g., "view1" and "view2") while may look different with more deformation when observed by two faraway views (e.g., "view1" and "view3"), so it is beneficial to develop a descriptor for the action from "view1" by capturing the "view1-view2" and "view1-view3" correlations.

For each action video, its mid-level correlation feature is represented by a set of decision values determined by the pre-learned classifiers of all the action classes from all the views. The pre-learned classifiers are trained using SVM classifiers in this work. Specifically, using each type of low-level visual feature, an independent SVM classifier is trained for each action class from each view. Based on seven types of low-level visual features mentioned in Section 4.1, seven independent SVMs are learned for each action class from each view to produce the decision values. Let us denote $f^l_{c,v}(x)$ as the pre-learned classifier of the c-th action class from the v-th view using the l-th type of visual feature extracted from action video x. Using the l-th type of visual feature, the likelihood that the video x belongs to the c-th action class captured by the v-th view is modeled by the classification score $g^l_{c,v} = f^l_{c,v}(x)$, and the corresponding correlation feature of x is then represented by $G^l = [g^l_{1,1}, ..., g^l_{C,1}, g^l_{1,2}, ..., g^l_{C,2}, ..., g^l_{C,V}]^{\mathrm{T}} \in \mathbb{R}^{\mathrm{D}}$, $\mathrm{D} = C \times V$, where C and V are the numbers of action classes and views, respectively. Similar to the low-level visual feature kernel, the mid-level correlation feature kernel is designed to be the combination of individual kernels:

$$K^{mid}_x(x_1, x_2) = \sum_{l=1}^{L} k^{mid}_l(x_1, x_2), \tag{10}$$

where $k^{mid}_l(x_1, x_2)$ is the kernel based on the l-th correlation feature and measures the similarity between x_1 and x_2 via the l-th correlation feature.

4.3 High-level Label Pair Kernel

The high-level action class-view label pair kernel K_u is expressed as

$$K_u(u_1, u_2) = \mathrm{J}(y_1, y_2) + \mathrm{J}(y_1, y_2) * \mathrm{J}(h_1, h_2), \tag{11}$$

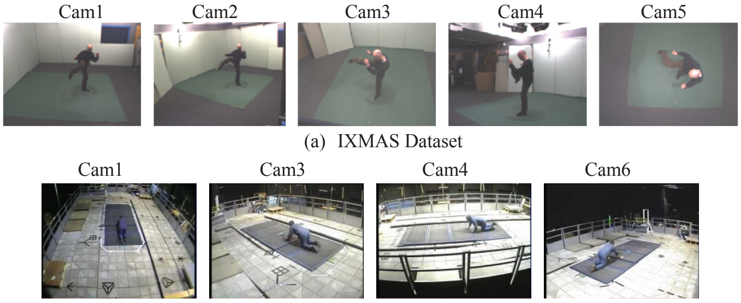
where y_i and h_i represent the labels of action and view, respectively. $u_i = (y_i, h_i)$ is the action class-view label pair. $\mathrm{J}(a, b)$ is the indicator function of $a = b$, namely, $\mathrm{J}(a, b) = 1$ if $a = b$, and $\mathrm{J}(a, b) = 0$ otherwise.

5 Experimental Results

5.1 Human Action Datasets

We evaluate the performance of our method and compare it with the state-of-the-art methods on two benchmark multi-view action datasets: IXMAS dataset and MuHAVi dataset. Fig.1 shows some action examples of these two datasets. The IXMAS dataset [8] consists of 12 complete action classes and each is executed three times by 12 subjects. Each action is recorded by five cameras observing the subjects from very different perspectives with the frame rate of 23fps and the frame size of 390×291 pixels. These actions are: check watch, cross arms, scratch head, sit down, get up, turn around, walk, wave, punch, kick, point and pick up. The body position and orientation are freely decided by different subjects.

The MuHAVi dataset [29] contains 17 human action classes: WalkTurnBack, RunStop, Punch, Kick, ShotGunCollapse, PullHeavyObject, PickupThrowObject, WalkFall, LookInCar, CrawlOnKnees, WaveArms, DrawGraffiti, JumpOverFence, DrunkWalk, ClimbLadder, SmashObject and JumpOverGap. Each action video is performed by seven actors and recorded using eight Schwan CCTV cameras with the frame rate of 25fps in a site with challenging lighting conditions provided by multiple sources of night street lights. Due to the computational complexity, we just choose the action videos captured by four cameras (i.e., two side cameras and corner cameras) in our experiment.



(a) IXMAS Dataset

(b) MuHAVi Dataset

Fig. 1. Sample frames from action videos on (a)IXMAS and (b)MuHAVi datasets

5.2 Experimental Setup

For interest points detection, the spatial and temporal scale parameters σ and τ are empirically set by $\sigma = 2$ and $\tau = 2.5$, respectively. 1000 interest points are extracted from each video and the size of cuboid around each point is empirically fixed as $7 \times 7 \times 5$. For the spatio-temporal context distribution feature of interest points, the number of space-time scales is fixed to 3. For the appearance feature of interest points, we first normalize the gray-level pixel values in each cuboid and then flatten each cuboid into a vector which is further reduced via Principle Component Analysis (PCA) by preserving 98% energy. Four descriptors of dense trajectory (i.e., trajectory, HOG, HOF and MBH) are extracted with the trajectory length of 15 and dense sampling step size of 5. The SIFT features are extracted from the 20% frames randomly selected from each video. We use the standard bag-of-words approach and construct a codebook for each visual descriptor separately and the number of visual words per descriptor is fixed to 2000.

For the visual feature kernel, we adopt the non-linear χ^2 kernel [24] defined by $k^{low}(x_1, x_2) = k_{\chi^2}(H_1, H_2) = exp(-\sum_{i=1}^{I} \frac{(h_{1i}-h_{2i})^2}{h_{1i}+h_{2i}})$, where $H_1 = \{h_{1i}\}$ and $H_2 = \{h_{2i}\}, i = 1, 2, ..., I$, are low-level visual features (i.e., the frequency histograms of word occurrences) of videos x_1 and x_2, respectively. I is the codebook size. For the correlation feature kernel, we use the non-linear RBF kernel expressed by $k^{mid}(x_1, x_2) = k_{\mathrm{RBF}}(d_1, d_2) = exp(-\frac{|d_1-d_2|^2}{2})$, where d_1 and d_2 are mid-level correlation features of videos x_1 and x_2, respectively.

5.3 Experimental Results

We learn the latent kernelized structural SVM (LKSSVM) model using the training videos from all the views without view labels and recognize the testing action from arbitrary single view. The leave-one-out cross validation strategy is employed in our experiment, in which videos of one subject are selected for testing and videos of the remaining subjects are used as training data.

1)Comparison with baseline methods: In order to evaluate the effectiveness of our method for view-invariant action recognition, we compare the recognition accuracies respectively using nonlinear SVM [30], latent structural SVM [15] and the proposed latent kernelized structural SVM on the low-level visual features. For the nonlinear SVM, χ^2 kernel is adopted and the one-against-all setting is applied to cope with multi-class classification task. For the latent structural SVM (LSSVM), the discriminative model is the same to that defined in Eq.(2). Different from LKSSVM using kernel functions in optimization, LSSVM formulates the model akin to conventional linear SVMs and the optimization problem is solved by the non-convex bundle algorithm proposed in [21].

Table 1 and Table 2 demonstrate the recognition results of different methods on the multi-view IXMAS dataset and MuHAVi dataset, respectively. It is interesting to have the observations as follows: 1) LKSSVM outperforms the baseline nonlinear SVM in terms of recognition accuracy on both datasets, which obviously demonstrates the benefit of modeling the view label as a latent variable and predicting it during both learning and inference. 2) By fusing multiple level information into a nonlinear kernel, LKSSVM achieves better results than LSSVM which formulates the model akin to conventional linear SVMs. The intuitive explanation is that kernel is able to encode complex relationships between two video-label pairs by evaluating the quality of the mutual matching between video-label pairs.

We additionally compare the performances using only low-level visual features, using only mid-level correlation feature and using the combined low-level visual features and mid-level correlation features, as shown in Table 3 and Table 4. The combination of low-level visual features and mid-level correlation features achieves the best results in all the cases, which demonstrates the effectiveness of using the decision values from the pre-learned classifiers of all the action classes from all the views to improve the recognition performance. Fig.2 illustrates the confusion table of recognition result on IXMAS dataset. It is interesting to observe that for some actions such as “sit down”, “walk” and “kick”, our method achieves very high recognition accuracies. Even for some challenging actions such as (e.g., “point”, “scratch head” and “wave”) that have small and ambiguous motions, our method still achieves reasonable and promising results. The confusion table of recognition result on MuHAVi dataset shown in Fig.3 also shows good performance of the proposed method for most actions.

Table 1. Accuracies (%) of different methods with visual features on IXMAS dataset.

Methods	View1	View2	View3	View4	View5	Ave.
SVM	88.89	84.03	84.72	81.25	78.47	83.47
LSSVM	87.50	83.33	86.11	81.25	86.11	84.86
LKSSVM	90.97	85.42	88.89	88.19	90.97	88.89

2)Comparison with other state-of-the-art methods: Different state-of-the-art methods may use different experimental settings. For a fair comparison, we conduct extensive experiments with the same experimental settings which are

Table 2. Accuracies (%) of different methods with visual features on MuHAVi dataset.

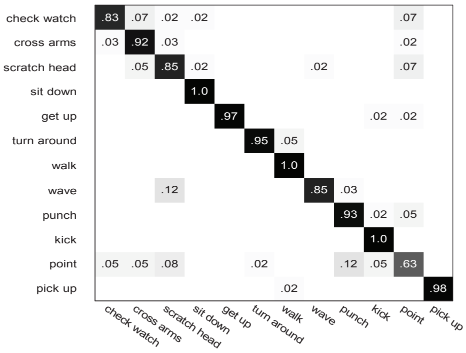
Methods	View1	View3	View4	View6	Ave.
SVM	93.28	92.44	93.28	95.80	93.70
LSSVM	91.60	94.12	95.80	95.80	94.33
LKSSVM	96.64	93.28	94.12	94.12	94.54

Table 3. Accuracies (%) using LKSSVM with different features on IXMAS dataset.

Features	View1	View2	View3	View4	View5	Ave.
Visual	90.97	85.42	88.89	88.19	90.97	88.89
Correlation	93.75	89.58	90.97	90.28	90.28	90.97
Visual+Correlation	94.44	90.97	91.67	89.58	88.89	91.11

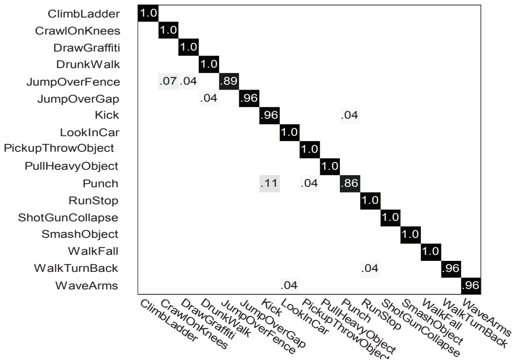
Table 4. Accuracies (%) using LKSSVM with different features on MuHAVi dataset.

Features	View1	View3	View4	View6	Ave.
Visual	96.64	93.28	94.12	94.12	94.54
Correlation	93.28	96.64	98.32	97.48	96.43
Visual+Correlation	96.64	97.48	98.32	97.48	97.48



	check watch	cross arms	scratch head	sit down	get up	turn around	walk	wave	punch	kick	point	pick up
check watch	.83	.07	.02	.02							.07	
cross arms	.03	.92	.03								.02	
scratch head		.05	.85	.02				.02			.07	
sit down				1.0								
get up					.97					.02	.02	
turn around						.95	.05					
walk							1.0					
wave			.12					.85	.03			
punch									.93	.02	.05	
kick										1.0		
point	.05	.05	.08			.02			.12	.05	.63	
pick up							.02					.98

Fig. 2. Confusion table of LKSSVM on IXMAS dataset.



	ClimbLadder	CrawlOnKnees	DrawGraffiti	DrunkWalk	JumpOverFence	JumpOverGap	Kick	LookInCar	PickupThrowObject	PullHeavyObject	Punch	RunStop	ShotGunCollapse	SmashObject	WalkFall	WalkTurnBack	WaveArms
ClimbLadder	1.0																
CrawlOnKnees		1.0															
DrawGraffiti			1.0														
DrunkWalk				1.0													
JumpOverFence		.07	.04		.89												
JumpOverGap				.04		.96											
Kick							.96				.04						
LookInCar								1.0									
PickupThrowObject									1.0								
PullHeavyObject										1.0							
Punch							.11		.04		.86						
RunStop												1.0					
ShotGunCollapse													1.0				
SmashObject														1.0			
WalkFall															1.0		
WalkTurnBack												.04				.96	
WaveArms								.04									.96

Fig. 3. Confusion table of LKSSVM on MuHAVi dataset.

applied in those state-of-the-art approaches. Table 5 illustrates the recognition performances of related methods when the video data from all views are used for training and the videos from one view are used for testing. In the setting with 11 actions, the "point" action is not considered. In the setting with 10 subjects, the "Pao" and "Srikumar" subjects are not considered. Compared with the methods [31] [8] [10], our method significantly improves the recognition performance in all five views with the same experimental setting. Owing to the fact that there are only 12 actions completely conducted by all the 12 subjects, we compare our method using 12 actions and 12 subjects with the methods [32] [33] using 13 actions and 12 subjects. Despite the slightly different setting, our method displays better results in all cases with the videos from four cameras (the top view excluded).

Moreover, a group of experiments are conducted, which test actions from one selected view while learn discriminative models from the other remaining views. There is no information from the testing view when learning the models. As shown in Table 6, our method significantly outperforms [12] for most views except the top view. The intuitive explanation is that the other four views are not able to provide enough information when testing on the top view since visual appearance of actions drastically varies from other four views to the top view. In [12], the connection between the top view and the other four views are learned during the training phase, which leads to the good recognition performance on the target top view.

Table 5. Accuracies (%) of state-of-the-art methods on IXMAS dataset. All these methods use the video data from all views for training and test on a single view. The columns "Act." and "Sub." respectively indicate the numbers of action classes and views.

Methods	Act.	Sub.	View1	View2	View3	View4	View5	Ave.
Weinland et al. [31]	11	10	86.7	89.9	86.4	87.6	66.4	83.4
Weinland et al. [8]	11	10	65.4	70.0	54.3	66.0	33.6	57.9
Junejo et al. [10]	11	10	74.8	74.5	74.8	70.6	61.2	71.2
Our method	11	10	98.18	97.27	98.18	95.45	96.36	97.09
Liu and Shah [32]	13	12	76.7	73.3	72.0	73.0	-	73.8
Reddy et al. [33]	13	12	69.6	69.2	62.0	65.1	-	66.5
Our method	12	12	95.14	89.58	91.67	90.28	-	91.67

6 Conclusions

We have proposed a novel latent kernelized structural SVM learning method for recognizing human actions from arbitrary views. Different from previous work on view-invariant recognition, we model the view label as a latent variable to address the difficulty of changing and unknown camera positions, which benefits the improvement of action recognition. In order to exploit the correlation between different view points and action classes, a novel mid-level correlation feature has been presented by using the decision values from pre-learned classifiers of

Table 6. Accuracies (%) of state-of-the-art methods on IXMAS dataset when one view is used for testing and the remaining views are used for training.

Methods	Act.	Sub.	View1	View2	View3	View4	View5	Ave.
Liu et al. [12]	11	12	86.6	81.1	80.1	83.6	82.8	82.8
Our method	11	12	92.42	95.45	93.18	87.12	62.88	86.21
Liu and Shah [32]	13	12	72.29	61.22	64.27	70.59	-	67.09
Reddy et al. [33]	13	12	81.0	70.9	79.2	64.9	-	74.0
Kaaniche and Bremond [34]	13	12	75.34	67.11	69.5	74.95	-	71.73
Our method	12	12	86.11	93.06	73.61	80.56	-	83.34

all the action classes from all the views. By designing a novel non-linear kernel function, we combine low-level visual cues, mid-level correlation features and high-level action class-view label pair information in a unified and principled framework. Extensive experiments on multi-view IXMAS and MuHAVi datasets have demonstrated that our method outperforms the state-of-the-art algorithms for view-invariant action recognition.

7 Acknowledgments

This work was supported by the Natural Science Foundation of China(NSFC) under Grant No. 90920009 and NSFC-Guangdong Joint Fund under Grant No. U1035004.

References

1. Gorelick, L., Blank, M., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. PAMI **29** (2007) 2247–2253
2. Yilmaz, A., Shah, M.: Actions sketch: a novel action representation. CVPR (2005)
3. Dollar, P., Rabaud, V., Cottrell, G., Belongie, S.: Behavior recognition via sparse spatio-temporal features. VS PETS (2005)
4. Niebles, J.C., Wang, H., Fei-fei, L.: Unsupervised learning of human action categories using spatial-temporal words. IJCV **79** (2008) 299–318
5. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local svm approach. ICPR (2004)
6. Yilmaz, A., Shah, M.: Recognizing human actions in videos acquired by uncalibrated moving cameras. ICCV (2005)
7. Shen, Y., Foroosh, H.: View-invariant action recognition using fundamental ratios. CVPR (2008)
8. Weinland, D., Boyer, E., Ronfard, R.: Action recognition from arbitrary views using 3d exemplars. ICCV (2007)
9. Yan, P., Khan, S.M., Shah, M.: Learning 4d action feature models for arbitrary view action recognition. CVPR (2008)
10. Junejo, I.N., Dexter, E., Laptev, I., Perez, P.: View-independent action recognition from temporal self-similarities. PAMI **33** (2011) 172–185

11. Lewandowski, M., Makris, D., Nebel, J.C.: View and style-independent action manifolds for human activity recognition. ECCV (2010)
12. Liu, J., Shah, M., Kuipers, B., Savarese, S.: Cross-view action recognition via view knowledge transfer. CVPR (2011)
13. Farhadi, A., Tabrizi, M.K.: Learning to recognize activities from the wrong view point. ECCV (2008)
14. Tsochantaridis, I., Hofmann, T., Joachims, T., Altun, Y.: Support vector machine learning for interdependent and structured output spaces. ICML (2004)
15. Yu, C.N.J., Joachims, T.: Learning structural svms with latent variables. ICML (2009)
16. Yu, C.N.J., Joachims, T.: Training structural svms with kernels using sampled cuts. ACM KDD (2008)
17. Wang, Y., Mori, G.: A discriminative latent model of object classes and attributes. ECCV (2010)
18. Wang, Y., Mori, G.: A discriminative latent model of image region and object tag correspondence. NIPS (2010)
19. Yang, W., Wang, Y., Mori, G.: Recognizing human actions from still images with latent poses. CVPR (2010)
20. Lan, T., Wang, Y., Yang, W., Mori, G.: Beyond actions: discriminative models for contextual group activities. NIPS (2010)
21. Artieres, T., Do, T.M.T.: Large margin training for hidden markov models with partially observed states. ICML (2009)
22. Zien, A., Ong, C.S.: Multiclass multiple kernel learning. ICML (2007)
23. Wu, X., Xu, D., Duan, L., Luo, J.: Action recognition using context and appearance distribution features. CVPR (2011)
24. Wang, H., Klaser, A., Schmid, C., Liu, C.L.: Action recognition by dense trajectories. CVPR (2011)
25. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV **60** (2004) 91–110
26. Xu, D., Chang, S.F.: Video event recognition using kernel methods with multilevel temporal alignment. PAMI **30** (2008) 1985–1997
27. Liu, J., Kuipers, B., Savarese, S.: Recognizing human actions by attributes. CVPR (2011)
28. Parikh, D., Grauman, K.: Relative attributes. ICCV (2011)
29. Singh, S., Velastin, S., Ragheb, H.: Muhavi: a multicamera human action video dataset for the evaluation of action recognition methods. AVSS (2010)
30. Chang, C.C., Lin, C.J.: Libsvm: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology (2001)
31. Weinland, D., Ozuysal, M., Fua, P.: Making action recognition robust to occlusions and viewpoint changes. ECCV (2010)
32. Liu, J., Shah, M.: Learning human actions via information maximization. CVPR (2008)
33. Reddy, K., Liu, J., Shah, M.: Incremental action recognition using feature-tree. ICCV (2009)
34. Kaaniche, M.B., Bremond, F.: Gesture recognition by learning local motion signatures. CVPR (2011)