Multi-View Action Recognition by Cross-domain Learning

Weizhi Nie 1, Anan Liu 1*, Jing Yu 1, Yuting Su 1, Lekha Chaisorn 2, Yongkang Wang 2, Mohan S Kankanhalli 2
1 School of Electronic Information Engineering, School of Computing, Tianjin University, China
2 National University of Singapore, Singapore

Abstract—This paper proposes a novel multi-view human action recognition method by discovering and sharing common knowledge among different video sets captured in multiple viewpoints. To our knowledge, we are the first to treat a specific view as target domain and the others as source domains and consequently formulate the multi-view action recognition into the cross-domain learning framework. First, the classic bag-of-visual word framework is implemented for visual feature extraction in individual viewpoints. Then, we propose a cross-domain learning method with block-wise weighted kernel function matrix to highlight the saliency components and consequently augment the discriminative ability of the model. Extensive experiments are implemented on IXMAS, the popular multi-view action dataset. The experimental results demonstrate that the proposed method can consistently outperform the state of the arts.

I. INTRODUCTION

Human action recognition has attracted increasing attention from both research and industry communities in recent years. It plays an important role in video surveillance, abnormal event system, human machine interaction. Many methods have been proposed in recent years [1][2][3][4][5]. Laptev et al.[6] extracted spatial-temporal features to represent human action and utilized SVM model to train classifier. Niebles et al.[7] proposed an unsupervised learning method for human action recognition, which applied topic model to find the correlation between different actions for recognizing. Recently, group sparse coding is used to extract high-level features to augment the discriminative ability of feature representation [8][9][10]. These methods have achieved recognition performance in some standard single-view human action datasets [11][12][13].

In more practical applications, multi-view cameras are always set to capture wider scenes. Multiple view cameras can capture more spatial information and more detail of action. For examples, in human rehabilitation system, it is expected that the automatic human action system can help patients to do standard rehabilitation action for rapid recovery. In this condition, one single camera does not capture enough information to judge the standard of action. Multi-view cameras can provide more detail and accurate information to help doctor to make the right decision. In the field of computer vision, more information can be used to train a more robust classifier for action recognition.

However, the feature representation of human action is sensitive to view point, it is difficult to leverage multi-view information for view-invariant model learning. To solve this problem, researchers must face two questions. The first question is that a robust descriptors is utilized to extract feature of human action. The second question is one excellent machine learning method can reduce the difference between human action features extracted from different views.

Different from previous work which only focusing on single-view action recognition [14] or learning and predicting from the source view and the target view respectively [15], we propose to explore a new way to take advantage of multi-view information for human action recognition. Specifically, we propose to consider the source view samples as the auxiliary data and integrate the classifiers trained on both domains to boost the performance in a cross-domain learning framework. First, we applied traditional spatial-temporal feature (STIP)[16] for interest point detection and description and then bag of visual word method is used for visual representation. Then, we integrated multi-view information in both target and source domains in the cross-domain learning framework to train a view-specific action model. The contributions are detailed below:

* Corresponding author Email: truman.nie@gmail.com

Fig. 1. Examples from IXMAS multiple view dataset. Each row shows one action viewed from different angles.
To our knowledge, we are the first to propose to leverage multi-view information for human action recognition in cross-domain framework;

- We propose a cross-domain learning method with the block-wise weighted kernel function matrix depending on the feature distributions among different domains to leverage multi-view information.

The rest of this paper is organized as follows: In section 2, we present the human action recognition methods under multiple views setting. In section 3, we will introduce the proposed cross-domain learning method in detail. The experimental results on IIXMAS dataset are presented in section 4. Section 5 concludes this paper.

II. RELATED WORKS

In recent years, many researchers have paid great attention to multiple views human action recognition[17][18]. Because human action are spatio-temporal patterns, there are two important problems, the spatio-temporal feature, and the modeling of robust visual patterns. According to these two problems, the current methods can be roughly divided into two categories.

For feature representation, many researchers are paing attention to the local interest point-based method. For volumetric feature extraction, the popular feature for spatio-temporal saliency is STIP[16], which using the distributions of the local gradients and optical flow by HoF[19] and HoG[20].

Rao et al.[14] proposed a novel action representation to capture the dramatic changes of action using spatio-temporal structure of 2D frame. This method requires reliable body joints detection and reliable tracking, which is still hard in computer vision. Besides, several methods[21][22][23][24][25] were proposed to apply 3D reconstruction for multiple views action recognition. However, 3D reconstruction or 3D structure feature requires reliable depth camera and strict coordinate of views, which is computationally expensive. Recently, Zheng et al.[15] proposed to extract high-level feature by sparse coding to reduce the feature difference among different views.

For model learning, in recent years, some researchers proposed to apply transfer learning method to address human action recognition problem. The goal of transfer learning method is to add auxiliary domain to train classifiers for target domain. However, traditional cross-domain learning method is based on the assumption that the target domain and auxiliary domain have the same distribution. This assumption is not realistic in multiple views human action recognition. Wang et al.[26] used the re-weighted method to redefine training samples for learning, which can reduce the feature gap between both domains. Raina et al.[27] applied self-taught learning method to find a new feature representation to improve target domain learning performance. Xu et al.[28] proposed a modified learning method, Domain Transfer SVM (DTSVM) inspired by A-SVM [25]. It utilizes the distance matrix between both domains as penal function for training classifier, which has a great performance in web video concept detection.

III. CROSS-DOMAIN LEARNING

Since the visual feature for human action is sensitive to view changes, simply adding action samples from other view as training samples can not effectively increase the performance, and might have negative influences on discriminative ability of the learned model. In order to to minimize the difference between the feature distribution from different domains and take advantage of multi-view information to improve the performance of action recognition in target view, we proposed a cross-domain learning method based on block-level weighted kernel function matrix. In the following sections, we will briefly review the A-SVM model[25] first and then present our method.

A. A-SVM Modeling

For A-SVM modeling, it is always assumed there are multiple auxiliary datasets \{\(D_1^a, D_2^a, \ldots, D_M^a\)\} with similar distribution to the target dataset \(D^t\). These auxiliary datasets can be trained to get multiple auxiliary classifiers \{\(f_1^a(x), f_2^a(x), \ldots, f_M^a(x)\)\}. These classifiers can be utilized to help target dataset \(D^p\) to generate a more robust classifier. The new decision function can be formulated as:

\[
f(x) = \sum_{k=1}^{M} t_k f_k^a(x) + \Delta f(x)
\]

\[
= \sum_{k=1}^{M} t_k f_k^a(x) + w^T \phi(x),
\]

where \(t_k \in (0, 1)\) is the weight of each auxiliary classifier \(f_k^a(x)\) and is constrained by \(\sum_{k=1}^{M} t_k = 1\). Based on the max-margin principle, the objective function is defined as:

\[
\min_{w} \frac{1}{2}||w||^2 + C \sum_{i=1}^{N} \xi_i
\]

\[
y_i \sum_{k=1}^{M} t_k f_k^a(x_i) + y_i w^T \phi(x_i) \geq 1 - \xi_i
\]

\[s.t. \quad \xi_i \geq 0,
\]

It can be formulated into the Lagrange dual form:

\[
L_D = \sum_{i=1}^{N} (1 - \lambda_i) \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j K(x_i, x_j),
\]

where \(\lambda_i = y_i \sum_{k=1}^{M} t_k f_k^a(x_i)\). Similarly, maximizing \(L_D\) under the KKT conditions gives the estimation of the model parameters \(\hat{a}\), and the adapted classifier is expressed as:

\[
f(x) = \sum_{k=1}^{M} \hat{a}_k f_k^a(x) + \sum_{i=1}^{N} \hat{a}_i y_i K(x, x_i).
\]

B. Our Approach

In A-SVM, \(f_k^a(x)\) is learned by one of the multiple auxiliary datasets. However, for action recognition task, there do not
exist many auxiliary datasets for model learning. Motivated by the MKL method [16], we leverage multiple kernel functions for model learning in auxiliary domain to increase the pre-learned classifiers. These pre-learned classifiers are used as prior information for learning a robust classifier for target domain, which can be formulated as \( \sum_{C=1}^{C} t_p f_p(x) \). We define the decision function in target domain as:

\[
f_T(x) = \sum_{c=1}^{C} t_p f_p(x) + \sum_{n=1}^{N} d_n w_n \phi_n(x) + b,
\]

where \( f_p(x) \) is a pre-learned classifier trained on the source-view dataset, \( \Delta f(x) = \sum_{n=1}^{N} d_n w_n \phi_n(x) + b \) is the perturbation function with the bias term \( b \). Based on the max-margin principle, the object function can be formulated as:

\[
\min_{w_m, \beta, \lambda, \xi_i} \left( \sum_{m=1}^{M} d_m |w_m|^2 + \lambda |\beta|^2 \right) + C \sum_{i=1}^{n} \xi_i,
\]

\[s.t. \ y_i f_T(x_i) \geq 1 - \xi_i, \quad \xi_i \geq 0,
\]

where, \( \beta = [\beta_1, ..., \beta_p] \) is the weight for each pre-learned classifier, \( \lambda, C \) are the regularization parameters. \( w_m \) is the parameters for the decision function. Eq.6 is not the standard SVM risk function. It is expected to transfer this risk function into one standard SVM risk function and apply SVM solver from libsvm [29] to solve this problem. Let \( \tilde{w} = [w_m', \sqrt{\lambda} \beta']' \) and \( \tilde{\phi}_m(x_i) = [\phi_m(x_i)', \sqrt{\lambda} \beta(\cdot)'(x_i)'] \), where \( f(x_i) = [f_1(x_i), ..., f_p(x_i)]' \), \( \tilde{w}_m = d_m \tilde{w}_m \). Eq.6 can be formulated into a quadratic programming problem:

\[
\min_{w_m, \beta, \lambda, \xi_i} \left( \sum_{m=1}^{M} \frac{1}{2} |\tilde{w}_m|^2 \right) + C \sum_{i=1}^{n} \xi_i,
\]

\[s.t. \ y_i \tilde{f}_m \tilde{\phi}_m(x_i) + b \geq 1 - \xi_i \quad \xi_i \geq 0.
\]

By the Lagrangian multipliers \( \alpha = [\alpha_1, ..., \alpha_n]' \), the dual of Eq.7 becomes:

\[
\min_{\alpha \in A} \frac{1}{2} (\alpha \circ y)' (\sum_{m=1}^{M} d_m \tilde{K}_m)(\alpha \circ y),
\]

where \( d_m \) is the weight of different kernel functions, \( \tilde{K}_m = [\tilde{\phi}_m(x_i) \tilde{\phi}_m(x_j)] \in R^{n_a \times n_a} \) is defined by the labeled training data from both domains, and \( \tilde{\phi}_m(x_i) \tilde{\phi}_m(x_j) = \phi_m(x_i) \phi_m(x_j) + \frac{1}{2} f(x_i)'f(x_j) \), where \( f(x_i) \) is the pre-learned classifier. In training process, the training samples can be selected from different domains. These samples should have different weight for the final decision. For example, if testing samples and training samples belong to the same domain, the related kernel function should have the bigger weight. With this assumption, we propose the block-wise weighted kernel matrix which can be formulated as:

\[
\tilde{K}_m = \begin{bmatrix} K_{m,A}^{A,A} & K_{m,A}^{A,T} \\ K_{m,T}^{T,A} & K_{m,T}^{T,T} \end{bmatrix} \rightarrow \begin{bmatrix} e_1 K_{m,A}^{A,A} & e_2 K_{m,A}^{A,T} \\ e_2 K_{m,T}^{T,A} & e_3 K_{m,T}^{T,T} \end{bmatrix}
\]

\[
K_{m,A}^{A} = [\tilde{\phi}_m(x_i) \tilde{\phi}_m(x_j)] \in R^{n_a \times n_a}, \quad x_i, x_j \in D^a
\]

\[
K_{m,A}^{T} = [\tilde{\phi}_m(x_i) \tilde{\phi}_m(x_j)] \in R^{n_a \times n_a}, \quad x_i \in D^a, x_j \in D^t
\]

\[
K_{m,T}^{A} = [\tilde{\phi}_m(x_i) \tilde{\phi}_m(x_j)] \in R^{n_a \times n_a}, \quad x_i \in D^t, x_j \in D^a
\]

\[
K_{m,T}^{T} = [\tilde{\phi}_m(x_i) \tilde{\phi}_m(x_j)] \in R^{n_a \times n_a}, \quad x_i, x_j \in D^t,
\]

where, \( n_a \) is the number of training samples from auxiliary domain, \( n_t \) is the number of training samples from target domain, \( D^a \) is the auxiliary dataset, \( D^t \) is the target dataset. Different weights of the kernel function matrix are defined as \( e_1 \) or \( e_2 \) according to the cross validation. Until now, the optimization problem in Eq.8 is in the same form as the dual form of SVM. Thus, it can be solved by existing SVM solvers [29].

C. Parameter Estimation

According to Eq.8, we need to compute \( d \) and \( \alpha \) for the final modeling. Following the coordinate descent theory, we first fix \( d \), and the dual variables \( \alpha \) can be solved by using SVM solvers. According to [30], \( d \) can be updated by:

\[
d_{t+1} = d_t - \zeta_t g_t,
\]

where \( g_t = (\nabla^2 G)^{-1} \nabla f G \) is updating direction and \( \zeta_t \) is the learning rate which can be obtained by [30]. Here, \( \nabla f G = \frac{1}{2}(\alpha \circ y) \), while \( \nabla^2 G = 0 \). We replace \( \nabla^2 G \) by \( \epsilon I \) to avoid no solution condition, where \( \epsilon \) is set as \( 10^{-5} \). Algorithm.1 shows the whole formulation:

**Algorithm 1 Modeling Training Process**

1: Initialization: \( d = \frac{1}{n_t} \).
2: for \( t = 1, 2, ..., T_{max} \) do
3:   Solve Eq.8 by SVM solvers to generate \( \alpha \).
4:   Update the \( d \) by Eq.10.
5: end for

By updating \( d \), we will get the optimal solution for the risk function Eq.8 and the corresponding classifier. Finally, we achieve the final action recognition model by integrating multi-view information.

IV. EXPERIMENTS

In our experiments, we compared the proposed method against several popular cross-domain learning methods, such as A-SVM and MKL. For performance evaluation, we use the Mean Average Precision (MAP) as in [31].

A. Data Set and Experimental Setup

We test the proposed method on the IXMAS multi-view action dataset [32] which contains eleven daily-live actions. Each action is performed three times by twelve action taken
from five different views. Fig.1 shows samples from the IXMAS action datasets.

In the experiment, we extract STIP [16] feature. Then, we apply K-means to quantize these interest point descriptors into 1,000 visual word as the vocabulary. With these basic view-dependent vocabularies, we conduct experiments on all possible pairwise view combinations to evaluate the proposed method. Based on the characteristic of our method, the leave-one-person-out strategy is utilized in our experiment. We selected only a few action samples from the target domain, and also selected the action samples from other views as auxiliary domains. Considering the experiment setting, the leave-one-person-out strategy is more suitable for evaluation. The details will be illustrated in the next section.

B. Learning with Pairwise Views

In this section, we evaluate our proposed method with pairwise views. IXMAS multi-view action dataset like Fig.1, it contains five views of 11 actions performed 3 times by 10 persons. We select pairwise views to test our approach. For examples, we selected action data from view 1 and view 2 respectively. We need to make twice experiments in this pairwise views. First, the action data from view 1 is used as target domain and that of view 2 is used as auxiliary domain. Only a few samples are selected from view 1, the target domain and all action samples of view 2 are selected for model learning. Second, the data of view 2 is used as target domain and that of view 1 is used as auxiliary domain to make the same experiment. Consequently, for IXMAS, we have 20 group experiments.

In the experiment process, we selected one person action data from target domain as the testing data and randomly selected other person action data from target domain combined with same action data from other view as training samples for model learning. The goal of experiment setting is to guarantee the size of training samples from target domain is less than the auxiliary domain, which will make the model learning even more dependent on auxiliary domain. This experiment setting is more suitable for evaluate the effective of the proposed method.

In our work, we used the parameter $e_1, e_2$ to change the weights of the kernel function value in training process. In order to demonstrate the important of this parameter, we first remove this parameter $e_1, e_2$ to learn classifier for each action, and than add this parameter to make the same experiment. The comparative experiments are implemented to demonstrate the correctness of the proposed method. The results without parameter $e_1, e_2$ are shown in the Table.I.

Table.I shows the experiment results by MAP. We observe that the results are very poor. Average precision is about 55% in IXMAS dataset. These results also demonstrate that action descriptor is sensitive for the view point. Thus, when we import the data from auxiliary domain as training samples, a bad result was generated. To make auxiliary domain useful for the model learning and improve the effective of the final classifier, we add the parameter $e_1, e_2$ to change the weights of kernel function. Here, we define $e_1 = 1.6$ and $e_2 = 1$, which is defined by statistical result. The goal of this definition is to increase weights of kernel function value from same domain and reduce the weight of kernel function from different domains. The new results are also shown in the Table.I.

From the new results, the new precisions obviously increase. In the training step, kernel matrix $K$ represents the key correlation between target domain and auxiliary domain. However, we did not ignore the usefulness of samples from same domain and should avoid the training samples of target domain are submerged by the large size of samples of auxiliary domain. Meanwhile, we did not ignore the correlated between different domains and save the parameter $e_2$. The experiment results demonstrate the effectiveness of our approach.

C. Comparison

To demonstrate the superiority of our proposed method, we compared against several popular cross-domains learning methods such as A-SVM and MKL. A-SVM is an popular cross-domain learning method, which has been used in many application such as: image concept detection, image annotation and so on. MKL is multiple kernel learning method, which applied labeled data from target domain and auxiliary domain to train classifier based on different kernel functions. We also compare our method with other baseline methods such as SVM-A, SMV-T and SVM-AT. SVM-A only uses auxiliary domain as the training data. SVM-T only uses target domain

<table>
<thead>
<tr>
<th>Methods</th>
<th>C0(T%)</th>
<th>C1(T%)</th>
<th>C2(T%)</th>
<th>C3(T%)</th>
<th>C4(T%)</th>
<th>Average(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Un-W Ours</td>
<td>* *</td>
<td>56.97</td>
<td>81.19</td>
<td>56.97</td>
<td>80.69</td>
<td>60</td>
</tr>
<tr>
<td>C0(S)</td>
<td>58.33</td>
<td>78.41</td>
<td>* *</td>
<td>59.40</td>
<td>79.21</td>
<td>* *</td>
</tr>
<tr>
<td>C1(S)</td>
<td>47.29</td>
<td>74.21</td>
<td>* *</td>
<td>48.04</td>
<td>75.36</td>
<td>* *</td>
</tr>
<tr>
<td>C2(S)</td>
<td>55.38</td>
<td>78.95</td>
<td>54.39</td>
<td>79.22</td>
<td>55.83</td>
<td>79.07</td>
</tr>
<tr>
<td>C3(S)</td>
<td>46.29</td>
<td>68.98</td>
<td>50.59</td>
<td>68.39</td>
<td>50.12</td>
<td>68.71</td>
</tr>
<tr>
<td>C4(S)</td>
<td>55.83</td>
<td>79.07</td>
<td>59.40</td>
<td>79.21</td>
<td>50.12</td>
<td>68.71</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Methods</th>
<th>C0(T%)</th>
<th>C1(T%)</th>
<th>C2(T%)</th>
<th>C3(T%)</th>
<th>C4(T%)</th>
<th>Average(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Un-W Ours</td>
<td>* *</td>
<td>56.97</td>
<td>81.19</td>
<td>56.97</td>
<td>80.69</td>
<td>60</td>
</tr>
<tr>
<td>C0(S)</td>
<td>58.33</td>
<td>78.41</td>
<td>* *</td>
<td>59.40</td>
<td>79.21</td>
<td>* *</td>
</tr>
<tr>
<td>C1(S)</td>
<td>47.29</td>
<td>74.21</td>
<td>* *</td>
<td>48.04</td>
<td>75.36</td>
<td>* *</td>
</tr>
<tr>
<td>C2(S)</td>
<td>55.38</td>
<td>78.95</td>
<td>54.39</td>
<td>79.22</td>
<td>55.83</td>
<td>79.07</td>
</tr>
<tr>
<td>C3(S)</td>
<td>46.29</td>
<td>68.98</td>
<td>50.59</td>
<td>68.39</td>
<td>50.12</td>
<td>68.71</td>
</tr>
<tr>
<td>C4(S)</td>
<td>55.83</td>
<td>79.07</td>
<td>59.40</td>
<td>79.21</td>
<td>50.12</td>
<td>68.71</td>
</tr>
</tbody>
</table>
as the training data. SVM-AT applies both domains as training samples. The experiment results are shown in the Table.II.

From the experiment results, we can find that SVM-AT works worst, while the results of SVM-T is better than those of SVM-AT, even better than the results of A-SVM and MKL. The reason is that the visual feature of the action video is sensitive to the changes of views. Thus, the auxiliary domain does not provide positive influence, while bringing a distraction for model learning. For each of five views, the proposed method achieves the best performance by effectively using the auxiliary domain and block-wise weighted kernel function matrix as well as reducing the mismatch in the data distributions between two domains. We can believe that the auxiliary domain with the kernel function matrix can well address multi-view human action recognition problem.

### D. Parameter Tuning

The parameter $e_1, e_2$ is the key point for kernal matrix computation. Different weights of kernel function decide the contributions of different samples for the final model. By changing the parameter $e_1$ and $e_2$, we hope to find the contribution of samples from both domains in the learning process. Fig.2 shows the final results.

![Fig. 2.](image-url)
From Fig.2, with the increasing of parameter $p$, average precision is rising. When $p = 1.6$, the average precision is at the peak. It means that the decision value is more dependent on the $K^{A,T}$ and $K^{T,A}$ values.

However, $K^{A,T}$ and $K^{T,A}$ are also important for the model learning. From Fig.2, with the continue increasing of parameter $p$, the important of $K^{A,T}$ and $K^{T,A}$ are reducing, and the average precision is also falling. The comparative experiment results demonstrate the effective of the proposed method.

V. CONCLUSION

In this paper, we propose a novel multi-view human action recognition method by utilizing common knowledge among different video sets captured in multiple viewpoints. We are the first to treat a captured view as target domain and the others as source domains and successfully convert the multi-view human action recognition into one cross-domain learning problem. In the model learning process, we propose a cross-domain learning method with block-wise weighted kernel function matrix to highlight the saliency components and improve the discriminative ability of the model. The experiment results clearly demonstrate the effectiveness of our method.

VI. ACKNOWLEDGEMENT

This work is partially supported by the National Natural Science Foundation of China(6110124, 21106095, 61107239, 61202168, 61303208), Specialized Research Fund for the Doctoral Program of Higher Education (20120032120023) and the grant of Elite Scholar Program of Tianjin University. Meanwhile, this research is partially supported by the Singapore NRF under its IRC@SG Funding Initiative and administered by the IDMPO.

REFERENCES