Mining Discriminative 3D Poselet for Cross-view Action Recognition

Jiang Wang[1] Xiaohan Nie[2] Yin Xia[1] Ying Wu[1]
[1]Northwestern University [2]University of California at Los Angles

Abstract

This paper presents a novel approach to cross-view action recognition. Traditional cross-view action recognition methods typically rely on local appearance/motion features. In this paper, we take advantage of the recent developments of depth cameras to build a more discriminative cross-view action representation. In this representation, an action is characterized by the spatio-temporal configuration of 3D Poselets, which are discriminatively discovered with a novel Poselet mining algorithm and can be detected with view-invariant 3D Poselet detectors. The Kinect skeleton is employed to facilitate the 3D Poselet mining and 3D Poselet detectors learning, but the recognition is solely based on 2D video input. Extensive experiments have demonstrated that this new action representation significantly improves the accuracy and robustness for cross-view action recognition.

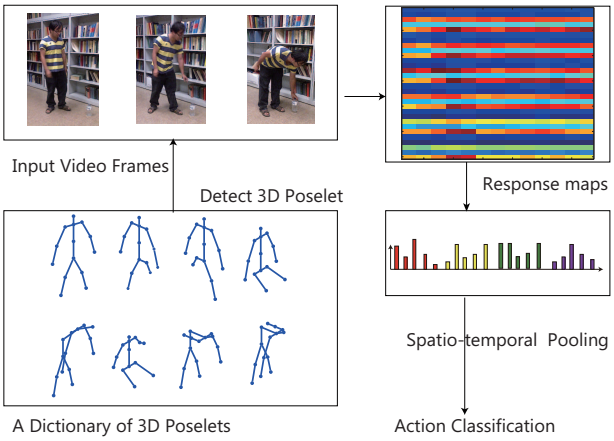


Figure 1. In 3D Poselet action recognition model, a set of view-invariant 3D Poselet detector is employed to detect the discriminative 3D Poselets, and the spatio-temporal pooling along a spatio-temporal pyramid is applied to roughly characterize the spatio-temporal configurations of the Poselet as feature vectors, which are utilized for action recognition.

1. Introduction

Cross-view action recognition methods recognize actions from views that are unseen in the training videos. This problem is a very challenging due to the large variations of the same action across different viewpoints. Despite some recent attempts [12, 7], it has not been well explored. The state-of-the-art approaches either *enumerate* a sufficiently large number of views and build dedicated feature and classifier for each view, or *interpolate* across views via transfer learning [12]. These methods require us to annotate a large number of videos for all views multiplied by all action categories.

We approach this problem from a novel perspective: creating a cross-view video action recognition representation with the spatio-temporal configuration of 3D Poselets. A 3D Poselet consists of a set of human parts with given appearance/motion features and specific 3D geometric relationship. For example, a 3D Poselet consisting of "head" and "right arm" parts as well as the movement of "right arm" towards the "head" is discriminative for the "drink" action. For each 3D Poselet, we learn a view-invariant 3D Poselet detectors. The 3D Poselet detectors are applied to all the frames of the input video to obtain a set of response maps, which are aggregated into pooled features with spatio-temporal pyramid pooling. The pooled features roughly characterize the spatio-temporal configuration of the Poselets, and are utilized as the features for recognition. Compared to local spatio-temporal feature-based approaches, the proposed 3D Poselet model can better captures the spatio-temporal structures of the actions. The framework of the proposed method is shown in Fig. 1.

The 3D Poselet representation is simple, but there are two major challenges in learning 3D Poselet representations. The first challenge is to find discriminative 3D Poselets. We design a novel data mining method to automatically discover the frequent and discriminative Poselet from the 3D Kinect human skeleton. This data-driven method provides a very effective way to discover discriminative 3D Poselets, while reducing the need of annotating the videos.

The second challenge is to learn view-invariant 3D Poselet detectors. This paper proposes a novel solution to address this difficult issue by learning the geometric relations among different views. To learn the multiple-view struc-

ture, we also take advantage of the 3D Kinect human skeleton as the 3D pose annotation. This 3D skeleton information is only available in training, but not used for cross-view action recognition. The projection of the 3D Poselets enables explicit modeling of geometric relationship in 2D views, and the appearances and motion are learned from the multiview training videos. Since 3D Poselet detectors share information across different views, training 3D Poselet detectors lead to better generalization than training a dedicated Poselet detector for each view.

3D Poselet model largely reduces the enormous demands on data annotation, while improving the accuracy and robustness of cross-view action recognition. We compare the 3D Poselet model with the state-of-the-art cross view action recognition methods in cross-subject, cross-view, and cross-environment settings. The experiments show that the 3D Poselet method not only outperforms the state-of-the-art cross view action recognition methods in all the three settings, but is also more robust to viewpoint and environment changes.

2. Related Work

Most of the cross-view action recognition algorithms are based on local visual features such as HOG [1] or HOF [9]. These methods either try to find the correlations of the local visual features across different viewpoints using transfer learning [11], or to find the local visual features that are robust to viewpoint changes, such as Hankelet [10] and self temporal similarity [7]. Hankelet [10] represents actions with the dynamics of short tracklets, and achieves cross-view action recognition by finding the Hankelets that are relatively invariant to viewpoint changes. Self temporal similarity [7] proposes to employ temporal self-similarities for cross-view action recognition. These methods work well on simple action classification, but they usually lack the discriminative power to deal with more complex actions.

Cross-view action recognition can also be achieved by modeling the 3D human skeleton. Recently, the development of depth cameras offers a cost-effective method to track 3D human skeleton [16]. Although the tracked 3D skeletons are noisy, it has been shown that they are useful to achieve good results in recognizing fine-grained actions [18]. View-invariant action recognition using 3D skeleton can be achieved by explicitly aligning the skeletons [17], or exploiting view-invariant canonical body poses and trajectories in 2D invariant space [14]. However, deploying a depth camera is not always possible in action recognition systems. As a result, we propose a method that only uses 3D skeleton as supervision in training, but does not need 3D skeletons inputs for action recognition in testing.

Recently, There is emerging interest in exploiting human poses for action recognition thanks to the steady progresses of single-image human pose estimation algorithms [19]. Yao et al. [20] estimates the 2D poses from the images, and matches the estimated poses with a set of representative poses. Desai et al. [2] learns a deformable part model (DPM) [3] that estimates both human poses and object locations. Maji et al. [13] uses the activation of *Poselets* detectors. Ikizler-Cinbis et al. [6] learns the pose classifier from web images for action recognition. In general, these methods were not specifically designed to handle cross-view actions. In contrast, this paper presents a new cross-view video action recognition approach based on human poses, inspired by the recent progress of multiview action recognition methods [4].

Part	Joints
Head	head, left shoulder, right shoulder
Torso	left hip, right hip, spine, left shoulder, right
Left leg	left hip, left knee, left ankle, left foot
Right leg	right hip, right knee, right ankle, right foot
Left arm	left shoulder, left elbow, left wrist, left hand
Right arm	right shoulder, right elbow, right wrist, right hand

Table 1. The relationship between joints and parts.

3. Mining Discriminative Poselet

3.1. The Human Part Configuration

We represent a human pose as a set of *human parts*, each of which consists of multiple human joints output by Kinect camera. We manually group the joints into six parts: "head", "torso", "left arm", "right arm", "left leg", and "right leg", as shown in Table 1. The human part acts as the basic building block for Poselet. A *Poselet* contains multiple human parts with specific configurations and geometric relationship.

Each joint i has a 3-dimensional location coordinates vector $\boldsymbol{p}_j(t) = (x_j(t), y_j(t), z_j(t))$, a 3-dimensional motion vector $\boldsymbol{m}_j(t) = (\Delta x_j(t), \Delta y_j(t), \Delta z_j(t))$, and a visibility label $h_j(t)$ at a frame t. $h_j(t) = 1$ denotes that the j-th joint is visible in frame t and $h_j(t) = 0$ otherwise. The human part configuration is characterized by its location, motion, and visibility vectors. Different actions can be characterized with the human parts with different configurations.

In order to make the human part configuration invariance to absolute body position, initial body orientation, and body size, we employ the up-right pose for orientation normalization. We find the frames where the human is approximately in an up-right pose, and use the poses of these frames for orientation alignment. If there is no up-right pose in a sequence, we do not perform orientation normalization. For each frame where the human subject is in an up-right pose, we fit a plane to the joints "head", "neck", "hip", "left shoulder", "right shoulder" with RANSAC procedure [5]. The

plane normal is used to align human orientation to an upright pose. We can also obtain the azimuth rotation angle $\theta(t)$ according to the orientation of the plane.

3.2. Part Clustering

Since the poses in one action are highly redundant, we cluster the examples of each human part to reduce the size of the search space, and to enable part sharing.

Let part k be one of the K parts of the person and $\mathcal{J}_k$ be the set of the joints in this part. For each joint $j \in \mathcal{J}_k$, we have its configuration $\boldsymbol{p}_j = (x_j, y_j, z_j)$, $\boldsymbol{m}(j) = (\Delta x, \Delta y, \Delta z)$, and $\boldsymbol{h}_i \in \{0, 1\}$ as its 3D position, 3D motion and visibility vectors, respectively. For a certain part, given the configurations of the two examples s and r, we can define their distance:

$$D_k(s,r) = \sum_{j \in \mathcal{J}_k} (\|\boldsymbol{p}_j^s(t) - \hat{\boldsymbol{S}}\boldsymbol{p}_j^r(t)\|_2^2 + \|\boldsymbol{m}_j^s(t) - \hat{\boldsymbol{S}}\boldsymbol{m}_j^r(t)\|_2^2)\,(1 + h_{s,r}(t)) \quad (1)$$

where $\hat{\boldsymbol{S}}$ is a similarity transformation matrix that minimizes the distance between the examples s and t for part k. The term $h_{s,r}$ is a penalty term based on the visibility of the joint j in the two examples:

$$h_{s,t}(j) = \begin{cases} a, & v_s(j) = v_r(j) \\ 0, & v_s(j) \neq v_r(j) \end{cases} \quad (2)$$

Since this distance is non-symmetric, we use the symmetric distance: $\bar{D}(s,r) = (D(s,r) + D(r,s))/2$.

Spectral clustering is performed on the distance matrix. Each cluster contains the human parts with the similar configuration and semantics, but they may be taken from different viewpoints. We remove the clusters that have too few examples, and use the rest of the clusters as the candidate part configurations for mining. The set of all candidates part configurations for the part k is denoted as: $\mathcal{T}_k = \{t_{1k}, t_{2k}, \cdots, t_{N_k k}\}$. Each t_{ik} is called a *part item*, and it is represented by the average joint position and motion vectors of the human part examples in the cluster.

3.3. Mining Representative and Discriminative Poselets

The discriminative power of a single part is usually limited. We need to discover Poselets (the combinations of the parts) that are discriminative for action recognition.

For a Poselet $\mathcal{P}$ that contains part item set $\mathcal{T}(\mathcal{P})$, where each part item in this set belonging to a different part. We define the configuration vector $\boldsymbol{p}_\mathcal{P}$ of a Poselet $\mathcal{P}$ as the 3D joint position and motion vectors of all the part items in the set $\mathcal{T}(\mathcal{P})$.

The activation of a Poselet $\mathcal{P}$ with configuration vector $\boldsymbol{p}_\mathcal{P}$ in a video v_i can be defined as:

$$a_p(i) = \min_t e^{-D(\boldsymbol{p}_\mathcal{P}, \boldsymbol{p}_\mathcal{P}^t)} \quad (3)$$

where t is the frame index of this video, $\boldsymbol{p}_\mathcal{P}^t$ is the configuration vector of the Poselet $\mathcal{P}$ in the t-th frame of video, and $D(.,.)$ is a distance function defined in Eq. (1). If very similar Poselet exists in this video, the activation is high. Otherwise, the activation is low. One discriminative Poselet should have large activation in the positive videos, while having low activation in the negative videos. We define the support of the Poselet $\mathcal{P}$ for category c as: $Supp_p(c) = \frac{\sum_{c_i=c} a_p(i)}{\sum_{c_i=c} 1}$ where c_i the category label of video v_i, and the discrimination of a Poselet p as: $\text{Disc}_p(c) = \frac{Supp_p(c)}{\sum_{c' \neq c} Supp_p(c')}$ We would like to discover the poses with large support and discrimination. Since adding one part item into a Poselet always creates another Poselet with lower support, i.e., $Supp_\mathcal{P}(c) < Supp_{\mathcal{P}'}(c)$ if $\mathcal{T}(\mathcal{P}) \supset \mathcal{T}(\mathcal{P}')$. Thus we can use the Aprior-like algorithm to find the discriminative Poselets. The detail of this algorithm can be found in Alg. 1.

For a Poselet $\mathcal{P}$, if there exist a Poselet $\mathcal{P}'$ such that $\mathcal{T}(\mathcal{P}) \subset \mathcal{T}(\mathcal{P}')$ and both $\mathcal{P}$ and $\mathcal{P}'$ are in the set of discriminative and representative Poselets, then $\mathcal{P}$ is a non-maximal Poseletx. In this algorithm, the non-maximal Poselets is removed from the discriminative Poselet pool.

Take a set of human parts $k \in \{1, 2, \cdots, K\}$, and their candidate part items $\mathcal{T}_k$, the number of classes C, threshold T_{supp} and T_{disc}.
for *Class* $c = 1$ *to* C **do**
  Set P_c, the discriminative Poselet pool for class c to be empty : $P_c = \{\}$. Set $k = 1$.
  repeat
    1) Generate the candidates Poselets by augmenting the Poselets in the pool P_C with the part items of the k-th part $\mathcal{T}_k$.
    2) Add the candidate Poselets whose support is larger than T_{supp} to the pool P_c.
    3) Remove the Poselets in P_c that is non-maximal.
    4) $k = k + 1$
  until *no discriminative Poselet is added to* P_c;
  remove the Poselets whose discriminative scores are less than T_{disc} in the pool P_c.
end
return the discriminative Poselets pool for all the classes.

Algorithm 1: Discriminative Poselet Mining

This algorithm usually produces an excessive large number of Poselets, we prune the Poselets with the following criteria:

- Remove Poselets that are similar to each other. This can be modeled as a set-covering problem, and can be solved with a greedy algorithm. We choose a Poselet $\mathcal{P}$

with highest discrimination, and remove the Poselets whose distance is less than a given threshold.

- Remove the Poselets with small validation scores for the detectors trained for these poses.

4. 3D Poselet Detector

For each Poselet, we train a viewpoint-robust 3D Poselet detector. To handle multi-view modeling, we learns a 3D deformable part model that characterizes the 3D geometric relationship of the *human parts* to a 3D Poselet projected to various views. One human part corresponds to a human part item in discriminative Poselet mining algorithm.

We use a star-shaped model for the dependencies among body parts, inspired by DPM [3], where the whole Poselet is represented as a root part, and the human parts are represented as the sub-parts. Fig. 2 shows an illustration of 3D Poselet detector model. Their 2D locations are denoted by $\mathcal{V} = \{\boldsymbol{v}_0, \boldsymbol{v}_1, \cdots, \boldsymbol{v}_N\}$, where $\boldsymbol{v}_0$ is for the root part (the whole pose). Denote by I the image frame. We define the detection score associated with the i-th *human part* to be $S_{\mathcal{R}}(\boldsymbol{v}_i, I, \theta)$ (details will be provided in Sec. 4.1).

Two factors contribute to the Poselet detection score: the detection score of its *parts* $S_{\mathcal{R}}(\boldsymbol{v}_i, I, \theta)$ and the spatial regularization among them $S_i(\boldsymbol{v}_0, \boldsymbol{v}_i, \theta)$, which specifies the geometric relationship of the root part and each child part. Such spatial regularization measures the compatibility among the parts from view θ (we only consider the rotation angle). In view of this, the total detection score of a Poselet is written as:

$$S_{\mathcal{V}}(\boldsymbol{v}_0, \boldsymbol{v}_1, \cdots, \boldsymbol{v}_N, \theta) = \sum_{i=0}^{N} S_{\mathcal{R}}(\boldsymbol{v}_i, I, \theta) + \sum_{i=1}^{N} S_i(\boldsymbol{v}_0, \boldsymbol{v}_i, \theta) \tag{4}$$

where $\boldsymbol{v}_i$ is the location of the part i, and θ is the view.

The 2D global location of a 2D Poselet is set to be the location of the root part, i.e.,$\boldsymbol{v}_0$. The Poselet detection score is computed by maximizing the detection scores over all possible projected views v_0:$S_{\mathcal{P}}(\boldsymbol{v}_0) = \max_\theta S_{\mathcal{V}}(\boldsymbol{v}_0, \theta)$

The evaluation of the spatial regularization of the parts needs a special treatment, because a Poselet represents a 3D Poselet and it can be projected to different views to lead to different part relationships explicitly, as illustrated in Fig. 2.

The 3D geometric relationship of the parts can be modeled as the 3D offsets of the i-th part with respect to the root part. Each offset can be modeled as a 3D Gaussian distribution with the mean $\boldsymbol{\mu}_i$ as well as diagonal covariance matrix $\boldsymbol{\Sigma}_i$. Here $\boldsymbol{\mu}_i$ can be estimated using the 3D skeleton data, and $\boldsymbol{\Sigma}_i$ will be learned (in Sec. 4.3).

The distribution of the 3D part offsets is projected to 2D for a given view. Here we assume scaled orthographic projections: $\boldsymbol{Q}_i^\theta$

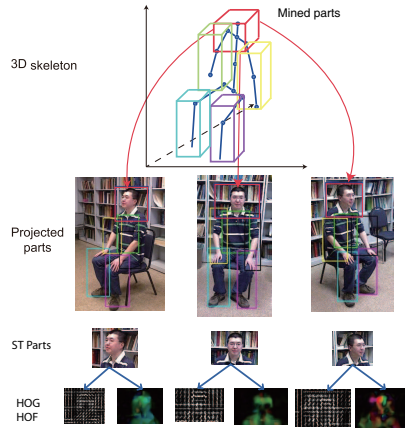


Figure 2. 3D parts and projected parts in different views.

$$\boldsymbol{Q}_i^\theta = \begin{bmatrix} k_1 \cos\theta & 0 & -k_1 \sin\theta \\ 0 & k_2 & 0 \end{bmatrix} \tag{5}$$

where θ is a rotation angle of the view, and k_1 and k_2 are the scale factors for two image axes. In training, we take advantage of the 3D skeleton data from Kinect cameras. Since we have the ground truth 3D (from 3D skeleton data) and 2D (from multiview videos) locations in our training data, these parameters can be easily estimated. The orthographic projection approximation works well in practice because the actors are usually sufficiently far away from the camera when performing actions. Since $\boldsymbol{Q}_i^\theta$ is a linear transform, the resulting projected 2D offset distribution is also a Gaussian distribution, with mean $\boldsymbol{\mu}_i^\theta = \boldsymbol{Q}_i^\theta \boldsymbol{v}_i$ and covariance matrix $\boldsymbol{\Sigma}_i^\theta = \boldsymbol{Q}_i^\theta \boldsymbol{\Sigma} (\boldsymbol{Q}_i^\theta)^T$. Thus the 2D spatial pairwise relationship score $S_i(\boldsymbol{v}_0, \boldsymbol{v}_i, \theta)$ can be written as follows:

$$\begin{aligned} S_i(\boldsymbol{v}_0, \boldsymbol{v}_i, \theta) = ((\boldsymbol{\Sigma}_i^\theta)_{11}^{-1}, (\boldsymbol{\Sigma}_i^\theta)_{22}^{-1}, (\boldsymbol{\Sigma}_i^\theta)_{12}^{-1})^T \cdot \\ (-\Delta u_i^2, -\Delta v_i^2, -2\Delta u_i \Delta v_i) \end{aligned} \tag{6}$$

where $(\Delta u_i, \Delta v_i) = \boldsymbol{v}_i - \boldsymbol{v}_0 - \boldsymbol{\mu}_i^\theta$ is the 2D deformation between the i-th part and the root part.

Since this 3D geometric relationship is shared and learned across different views, the 2D geometric relationship of the novel views can be obtained by projecting the 3D geometric relationship to the novel views.

4.1. Motion/Appearance

The spatio-temporal patterns of a human part under a view are modeled as its motion and appearance features. Each *part* has an *appearance* score $A_i(\boldsymbol{v}_i, I, \theta)$, and a *motion* score $M_i(\boldsymbol{v}_i, I, \theta)$. They capture the likelihood (or compatibility) of the appearance and motion of the part i located at location $\boldsymbol{v}_i$ under view θ, respectively. The score associated with a *human part* is thus written as:$S_{\mathcal{R}}(\boldsymbol{v}_i, I, \theta) = A_i(\boldsymbol{v}_i, I, \theta) + M_i(\boldsymbol{v}_i, I, \theta)$.

We exploit commonly used HOG [1] and HOF [9] features to represent the appearance and motion, respectively. In order to model the difference and correlation of the appearance and motion for one part in different view, we discretize the view angle θ into M discrete bins (each bin corresponds to a view node), and use exponential interpolation to obtain the appearance and motion features in the novel view bins. The appearance score function $A_i(\boldsymbol{v}_i, I, \theta)$ and are defined as

$$A_i(\boldsymbol{v}_i, I, \theta) = \frac{\sum_{m=1}^{M} e^{-d^2(\theta,\theta_m)} \boldsymbol{\phi}_{i,m}^T \phi(I, \boldsymbol{v}_i, \theta)}{\sum_{m=1}^{M} \alpha_m} \quad (7)$$

where $e^{-d^2(\theta,\theta_m)}$ is the exponential of angular distance between the view θ and the view of bin m, $\boldsymbol{\phi}(I, \boldsymbol{v}_i, \theta)$ is HOG feature at the location $\boldsymbol{v}_i$ in image I under the view θ. $\boldsymbol{\phi}_{i,m}$ is the HOG template of view bin m, and need to be learned from the training data (see Sec. 4.3). The motion score function $M_i(\boldsymbol{v}_i, I, \theta)$ is defined in a similar way.

Thus, the appearance and motion of human part are shared across different views, and we can learn the appearance/motion of the part nodes for the novel views via interpolation.

4.2. Inference

Given an input video from a novel view, the inference of 3D Poselet detector calculates the scores of all the 3D Poselet detectors projected to all the possible views. The highest detection score among all the views is utilized as the 3D Poselet detection score. Since this 3D Poselet model is tree-structured, inference can be done via dynamic programming that is similar to DPM inference algorithm [3].

We apply the spatio-temporal pyramid to represent the spatio-temporal relationship of 3D Poselet for action recognition. The detection scores of the 3D Poselet detectors at different locations and frames constitute a sequence of response maps. We apply the max-pooling over a spatio-temporal pyramid. The response of a cell in the pyramid is the maximum among all responses in this cell.

We divide one whole video into 3-level pyramid in the spatio-temporal dimensions. This yields $1+8+64 = 73$ dimensional feature vector for each response map. Then, We train a linear SVM on these features for action classification. Although this representation only acts as a rough description of the spatial-temporal relationships between the poses, we find it achieves very good results on our experiments.

4.3. Learning

Learning 3D Poselet detector parameters can be formulated as a latent structural SVM problem. The parameters of the latent SVM include: the variance $\boldsymbol{\Sigma}_i$, the appearance and motion templates $\boldsymbol{\beta}_{i,m}$ and $\boldsymbol{\gamma}_{i,m}$ in Eq. (7).

Although we have the non-root part locations and the view available in the training data, we treat the locations of the parts $\boldsymbol{v}_j$ and the view θ as latent variables and the labeled location of the parts and the view angle are used as initialization., because we are more interested in predicting the position of the Poselet rather than the precise location of each part and the view. This treatment is more robust to the noises in the part locations in the training data.

The learning algorithm is similar to DPM learning algorithm [3], and is done by iterating between optimizing $\beta, \gamma, \boldsymbol{\Sigma}_i$, and calculating the part locations and the views of the positive training data.

For each 3D Poselet, we use the samples whose distances are less than η to this Poselet in the positive videos as positive examples, and randomly sample 5000 negative training examples from negative videos. We apply two bootstrapping mining of hard negatives during the learning process. As the prediction score is a linear function of the parameters $\boldsymbol{w}_i$, the parameters $\boldsymbol{w}_i$ can be easily learned via a linear SVM solver.

5. Experiments

5.1. Northwestern-UCLA Multiview Action 3D

The Northwestern-UCLA Multiview Action 3D dataset[1] contains RGB, depth and human skeleton data captured simultaneously by three Kinect cameras. This dataset include 10 action categories: *pick up with one hand*, *pick up with two hands*, *drop trash*, *walk around*, *sit down*, *stand up*, *donning*, *doffing*, *throw*, *carry*.

The comparison of the recognition accuracy of the proposed algorithm with the baseline algorithms is shown in Table 2. We compare with virtual views [11], Hankelet [10], Action Bank [15] and Poselet [13]. For Action Bank, we use the actions provided by [15] as well as a portion of the videos in our dataset as action banks. For Poselet, we use the Poselets provided by [13]. We also compare our model with training one dedicated deformable part model for each view, which is essentially a mixture of deformable part models (DPM), to compare the robustness of the proposed method under different viewpoints with DPM model. We have 50 *3D Poselet* and 10 view bins for every action. These parameters are chosen via cross-validation. In 3D Poselet model, the appearance/motion information and geometrical relationship of the *parts* are shared and learned across different *view*, but the mixture of DPM treats them independently.

We perform recognition experiments under three settings: 1) *cross-subject* setting: We use the samples from 9 subjects as training data, and leave out the samples from 1 subject as testing data. 2) *cross-view* setting: We use the

[1] http://users.eecs.northwestern.edu/∼jwa368/my_data.html

Method	C-Subject	C-View	C-Env
Virtual View [11]	0.507	0.478	0.274
Hankelet [10]	0.542	0.452	0.286
Action Bank [15]	0.246	0.176	N/A
Poselet [13]	0.549	0.245	0.485
Mixture of DPM	0.748	0.461	0.688
3D Poselet	**0.816**	**0.733**	**0.793**

Table 2. Recognition accuracy on Multiview-3D dataset.

samples from 2 cameras as training data, and use the samples from 1 camera as testing data. 3) *cross-environment* setting: We apply the learned model to the same action but captured in a different environment. These settings can evaluate the robustness to the variations in different subjects, from different views, and in different environments.

The proposed algorithm achieves the best performance under all three settings. Moreover, the proposed method is very robust under the cross-view setting. In contrast, although the state-of-the-art local-feature-based cross-view action recognition methods [10, 11] are relatively robust to viewpoint changes, the overall accuracy of these methods is not very high, because the local features are not enough to discriminate the subtle differences of the actions in this dataset. Moreover, these methods are sensitive to the changes of the environment. The Poselet method is robust to environment changes, but it is sensitive to viewpoint changes. Since the mixture of DPM does not explicitly model the relations across different view, its performance degrades significantly under cross-view setting.

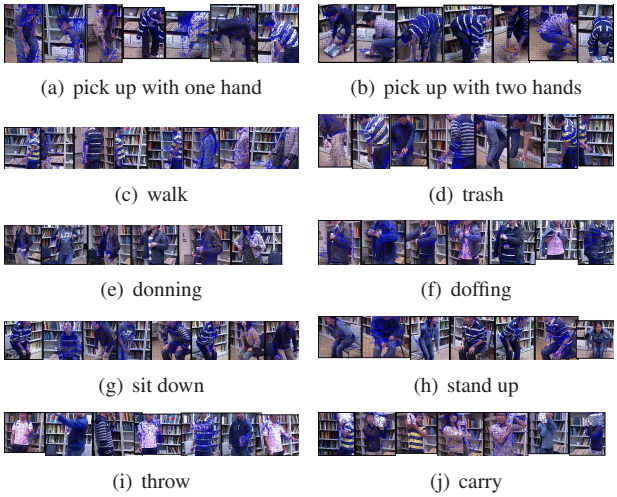
Some examples of the discriminative 3D Poselet discovered with the proposed data mining algorithm is shown in Fig. 3. One subfigure in this figure corresponds to a discriminative Poselet for a specific action. We can see that one Poselet contains frames that are captured from different viewpoints and performed by different subjects. As a result, the frames for one 3D Poselet can differ significantly in their appearance and motion. However, the proposed 3D Poselet model can successfully represent the spatio-temporal patterns of one Poselet across different viewpoints.

6. Conclusion

We propose a new cross-view action representation based on mining 3D Poselet, which can effectively express the geometry, appearance and motion variations of actions across multiple view points. It takes advantage of 3D skeleton data to train, and achieves 2D video action recognition from unknown views. Our extensive experiments have demonstrated that 3D Poselet model significantly improves the accuracy and robustness for cross-view, cross-subject and cross-environment action recognition.

References

[1] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *CVPR*, pages 886–893. IEEE, 2005.



(a) pick up with one hand (b) pick up with two hands (c) walk (d) trash (e) donning (f) doffing (g) sit down (h) stand up (i) throw (j) carry

Figure 3. Samples of the discovered discriminative poses. Each row shows a discriminative Poselet for one class, captured from different viewpoints. We can observe that one Poselet captured from different viewpoints have huge difference in the appearance and motion.

[2] C. Desai and D. Ramanan. Detecting Actions , Poses , and Objects with Relational Phraselets. In *ECCV*, 2012.

[3] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *PAMI*, 32(9):1627–45, Sept. 2010.

[4] S. Fidler, S. Dickinson, and R. Urtasun. 3D Object Detection and Viewpoint Estimation with a Deformable 3D Cuboid Model. In *NIPS*, pages 1–9, 2012.

[5] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[6] N. Ikizler-Cinbis and S. Sclaroff. Web-Based Classifiers for Human Action Recognition. *Multimedia, IEEE Transactions on*, 14(4):1031–1045, 2012.

[7] I. N. Junejo, E. Dexter, I. Laptev, and P. Patrick. Cross-View Action Recognition from Temporal Self-Similarities. In *ECCV*, 2008.

[8] I. Laptev. On Space-Time Interest Points. *IJCV*, 64(2-3):107–123, Sept. 2005.

[9] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, pages 1–8, 2008.

[10] B. Li, O. I. Camps, and M. Sznaier. Cross-view activity recognition using hankelets. In *CVPR*, pages 1362–1369. IEEE, 2012.

[11] R. Li and T. Zickler. Discriminative virtual views for cross-view action recognition. In *CVPR*, pages 2855–2862. Ieee, June 2012.

[12] J. Liu, M. Shah, B. Kuipers, and S. Savarese. Cross-view action recognition via view knowledge transfer. In *CVPR*, pages 3209–3216. Ieee, June 2011.

[13] S. Maji, L. Bourdev, and J. Malik. Action recognition from a distributed representation of pose and appearance. In *CVPR*. IEEE, June 2011.

[14] V. Parameswaran and R. Chellappa. View invariance for human action recognition. *IJCV*, 66(1):83–101, 2006.

[15] S. Sadanand and J. Corso. Action bank: A high-level representation of activity in video. In *CVPR*, number May, 2012.

[16] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *CVPR*, 2011.

[17] J. Wang and M. Elad. Learning Actionlet Ensemble for 3D Human Action Recognition. *PAMI*, pages 76–88, October 2013.

[18] J. Wang, Z. Liu, Y. Wu, and J. Yuan. Mining Actionlet Ensemble for Action Recognition with Depth Cameras. In *CVPR*, 2012.

[19] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts resenting shape. *CVPR*, 2011.

[20] B. Yao and F.-F. Li. Action Recognition with Exemplar Based 2.5D Graph Matching. In *ECCV*, 2012.