Hierarchical 3D Kernel Descriptors for Action Recognition Using Depth Sequences

Yu Kong, Behnam Satarboroujeni, and Yun Fu

1 Department of Electrical and Computer Engineering, 2 College of Computer and Information Science.
Northeastern University, Boston, MA, USA

Abstract—Action recognition is a challenging task due to intra-class motion variation caused by diverse style and duration in performed action videos. Previous works on action recognition task are more focused on hand-crafted features, treat different sources of information independently, and simply combine them before classification. In this paper we study action recognition from depth sequences captured by RGB-D cameras using kernel descriptors. Kernel descriptors provide an elegant way for combining a variety of information sources and can be easily applied to a hierarchical structure. We show how using kernel descriptors over pixel-level attributes in video sequences gains a great success compared to state-of-the-art methods. Following the success of kernel descriptors [1] on object recognition tasks, we employ 3D kernel descriptors, which are a unified framework for capturing pixel-level attributes and turning them into discriminative low-level features on individual 3D patches. We use efficient match kernel (EMK) [2] as the next level of our hierarchical structure to abstract the mid-level features for classification. Through extensive experiments we demonstrate using pixel-level attributes in the hierarchical architecture of our 3D kernel descriptor and EMK achieves superior performance on the standard depth sequences benchmarks.

I. INTRODUCTION

Despite many research efforts in recognizing human actions and many encouraging advances in this field, accurate action recognition is still a challenging task because of large intra-class motion variation. Introducing low-cost devices such as Kinect sensors has triggered many research activities for achieving concise descriptions in recognition tasks due to their availability of depth sequences alongside the RGB data. Insensitivity of depth images to different lighting situations and illuminations is an effective advantage compared to color images. Moreover, depth sequences provide additional shape and movement information due to providing accurate distance information for each pixel in images.

In using depth sequences, recent works such as [3] consider human motion as a posture of body segments, and employ skeleton trackers to achieve discriminative representations from depth sequences. However, as presented in [4], utilizing low-level attributes in depth images outperforms recent high-level representations with improvement on capturing joint shape-motion cues. This idea leads us towards employing low-level attributes in depth images in a more elaborate way to capture accurate information in describing an action scene.

Depth sequences [5], [4], [6] are noisy with undefined depth data and incorrect joint data. Existing work [3], [4], [6] directly builds on low-level noisy features. However, these noisy data of different categories are hardly being linearly separated, and thus would hurt the performance. In addition, the correlations between human body parts are highly nonlinear. It is difficult to model their joint distribution accurately by extracting features from each of them and concatenating the two features.

In this work, we propose a hierarchical kernel-based method to learn non-linear correlations between RGB and depth action data for action recognition. The aforementioned problems are overcome by a novel hierarchical kernel framework motivated by the recent success of kernel descriptors in pixel-level attributes for object recognition task [1]. We present a 3D gradient kernel descriptor which is a low-level depth sequence descriptor with the ability of capturing detailed information by computing pixel-level 3D gradient. Our 3D gradient kernel is intuitively producing normal vectors on the surface of 3D geometric shape of scene using the gradient in depth images. Moreover, the gradient information is computed along the temporal dimension as well as the spatial dimensions to describe the change in shape of the 3D surface in time. As it was shown in [7], using the normal
vectors in depth images provides a rich description of the scene.

At higher level, we employ the efficient match kernel (EMK) to summarize mid-level features for classification. The use of EMK allows us to learn nonlinear correlations accurately measure the similarities between two RGB-RGB-DD action videos, and provide rich mid-level information to bridge the semantic gap for classification.

The main contribution of this paper is employing kernel descriptors for converting pixel-level 3D gradient in depth sequences into patch-level features. Next step is to employ EMK [2], which is a kernel representation of bag-of-words method, to build a hierarchical structure upon the low-level patch features to produce mid-level feature vector (Fig. 1). Finally, the classification task is achieved by using linear SVM. We show how this framework is applied to depth sequences for achieving discriminative information and surpassing sophisticated learning approaches based on high-level features. We extensively evaluate our method on standard depth sequence datasets, MSR Action 3D [5], MSR Action Pairs [4], and MSR Gesture 3D datasets [6], and achieve superior performance over state-of-the-art methods.

II. RELATED WORK

Action recognition has been widely studied in the computer vision community. Previous methods mainly recognize actions from color videos. A majority of existing methods classify actions directly based on low-level features extracted from color videos. For example, methods proposed in [8], [9], [10], [11], [12] extract low-level features such as spatio-temporal interest points, key poses [15], optical flow [18], structure information [14], and body shape feature [13], etc., from action videos, and use classifiers to recognize actions.

Though effective, these methods heavily rely on hand-crafted features. To overcome this problem, deep learning techniques [16] have been introduced recently to learn action features from videos.

In spite of having promising results in action recognition, these studies do not present promising action representations due to large semantic gap. Mid-level feature based methods are proposed to address this problem. For example, motion attributes [17], interactive phrases [19], and action context [20], are proposed to better summarize low-level features. The learned mid-level features are effective in reducing noise in low-level features, and extracting discriminative information from them, and thus are able to boost recognition performance.

Researchers have paid much attentions to action recognition from RGB-D data [21], [3], [22], [23], [24], [4] captured by RGB-D sensors such as the cost-effective Kinect sensors. Compared with conventional color videos, the additional depth information in RGB-D data enables us to capture 3D geometric information of the scene, which can help remove background noise and simplify intra-class motion variations. Previous works [25] also tried to use methods to reconstruct the depth information in datasets with just RGB data. Some of early researches on RGB-D data were based on treating the additional depth information as just another type of 2D information. Employing old descriptors on depth data and using the extracted features alongside the RGB features have been explored [24]. Although some methods such as [26] can extract additional information specifically for depth data. Among these methods, as demonstrated in [7], [27] surface normal provides a rich set of shape and structure information in depth sequences. HOND4D [4] followed the same method with extending the surface normals to a 4D space and quantized them to achieve discriminative information about the scene in this space.

Our work also uses the concept of surface normal and employs it alongside with kernel descriptors to eliminate the loss of information in quantization part. Kernel descriptors are easy to design and can outperform other methods which are built upon quantization of continuous data with gathering more aspects of the input data for accurate presentation in the learning phase.

III. OUR METHOD

Given a sequence of depth images we design a match kernel for gathering pixel-level gradient information which is equivalent to surface normals [7] in 3D geometric view. Following the methodology in [1] we learn compact basis vectors using KPCA [28] and then we build our kernel descriptor with projecting infinite-dimension feature vectors which are built upon our 3D gradient match kernel, into this finite set of basis vectors.

A. Kernel Descriptors

Kernel descriptors have been demonstrated to be very effective in capturing pixel-level features compared to other methods such as SIFT [29] and HOG [30], as they are able to gather more descriptive information lying in high dimensional space. The main contribution of this paper is presenting 3 dimensional kernel descriptor over normal vectors on the 3D depth surface with expanding the current kernel descriptors for object recognition task in 2D images.

Based on the description in [1], 3D gradient match kernel

$$K_{3D}(P, Q) = \sum_{z \in P} \sum_{z' \in Q} \tilde{m}_{z} \tilde{m}_{z'} k_p(\theta_z, \theta_{z'}) k_p(z, z')$$

(1)

is the kernel representation of orientation histograms over video patches, where $P$ and $Q$ are the 3D patches from different videos and the overall output is a measurement of similarity between them. Let $m_z$ and $\theta_z$ be the 3D magnitude and orientation of gradient at the pixel $z$ in a patch, then $
\tilde{m}_{z} = m_z / \sqrt{\sum_{z \in P} m(z)^2 + \epsilon}$ is the normalized magnitude of gradient which is used as a weight factor. Gaussian orientation kernel $k_o(\theta, \theta') = \exp(-\gamma_o ||\theta - \theta'||^2) = \phi_o(\theta_z)^\top \phi_o(\theta_{z'})$ is for measuring similarity between orientations of gradient for corresponding pixels, and similarly $k_p(z, z') = \exp(-\gamma_p ||z - z'||^2) = \phi_p(z)^\top \phi_p(z')$ is the Gaussian kernel for measuring how close each pair of pixels are in 3D spatio-temporal dimension of each patch.
In case of depth sequences, we have a richer description in each frame compared to RGB images. Because alongside of spatial location of each pixel, we are also given a third dimension which is the distance of each pixel from camera, shown as the intensity value. This fact helps us to capture the shape of 3D geometric surface by considering the orientation and magnitude of 2D gradient in each pixel of a depth map. Using 3D gradient alongside of spatio-temporal dimension we are capturing the change of shape of this surface over time, which is an essential factor for having a rich discriminative representation of each 3D patch.

B. Feature Extraction

The 3D gradient match kernel uses two patches and gives a measurement of similarities between them. However, the goal of kernel descriptors is to produce an independent discrimination over each individual patch. With expanding the orientation and position kernels in equation (1), the extracted features over each patch is represented as

\[ F_{3D}(P) = \sum_{z \in P} \tilde{m}(z) \phi_o(\theta(z)) \otimes \phi_p(z), \]  

(2)

where \( \otimes \) is the Kronecker product. Instead of measuring the similarities between two patches, \( F_{3D}(P) \) is only depended on one patch and is used as the feature vector. Extracting the output of equation (1) over patches \( P \) and \( Q \) is done by \( K_{3D}(P, Q) = F_{3D}(P)^T F_{3D}(Q) \). However, the goal of kernel descriptors is to extract individual features based on \( F_{3D}(P) \). Presence of Gaussian kernel in formulation of \( F_{3D}(P) \) makes it to have infinite dimensionality. Therefore, computing \( F_{3D}(P) \) is infeasible. For reducing the dimensionality we project the \( F_{3D}(P) \) to a finite set of basis vectors. In choosing an efficient set of basis vectors following the presented method in [2], we use a fine grid over the support region for approximation. Let \( \{ \phi_o(x_i) \}_{i=1}^b \) be the set of basis vectors where \( b \) is the number of basis vectors and \( x_i \) is the sample normalized vector used for approximation of Gaussian kernel \( \tilde{k}_p(z, z') \) over position of pixels in 3D space. The mechanism of projecting infinite dimension vector \( \phi_o(z) \) to the low-dimensional basis vector set \( \{ \phi_p(x_i) \}_{i=1}^b \) is equivalent to using the following finite dimensional kernel:

\[ \tilde{k}_p(z, z') = k_p(z, X)^T [K_p^{-1}]_{ij} k_p(z', X) \],

(3)

where \( k_p(z, X) = \{[k_p(x_i, x_1), \ldots, k_p(x_i, x_b)] \}^T \) is a vector with size equal to number of basis vectors, \( K_p \) is a square matrix with \( b \) dimensions, and \( K_p^{-1} = G^T G \).

With following equation (3) and writing the same for orientation kernel with \( b_o \) basis vectors, we have finite dimension feature vector \( \tilde{F}_{3D}(P) = \sum_{z \in P} \tilde{m}(z) \phi_o(\theta(z)) \otimes \phi_p(z) \), where \( \phi_o(\theta(z)) = G k_o(\theta, X) \) with only \( b_o \) dimension and \( \phi_p(z) = G k_p(z, X) \) with \( b_p \) dimension.

C. Dimensionality Reduction

Presence of Kronecker product in producing the feature vector alongside with using grid approximation on 3D space make \( \tilde{F}_{3D}(P) \) to have a high dimensionality. In particular, we quantize the position kernel \( k_p \) with basis vectors on a \( 4 \times 4 \times 4 \) grid, and gradient orientation kernel \( k_o \) with basis vectors on a \( 6 \times 6 \times 6 \) grid in all experiments. Therefore, the final dimensionality of \( \tilde{F}_{3D}(P) \) is \( 64 \times 216 = 13,824 \). Although we project \( F_{3D}(P) \) to finite dimension now, the dimensionality is still too high for empirical use.

For dealing with aforementioned problem and handling the computation cost, we use the formulation in [1] and try to project our feature vector to a set of joint basis vectors \( \{ \phi_o(x_i) \otimes \phi_p(y_j, z_l) \} \), where \( \{ \phi_o(x_i) \}_{i=1}^{b_o} \) and \( \{ \phi_p(y_j) \}_{j=1}^{b_p} \) are the set of basis vectors of orientation and position kernels approximation accordingly.

D. Hierarchical Kernel Descriptors

Our 3D kernel descriptor produces low-level discriminative features over each 3D patch in video. Fig. 2 shows the representation of feature vectors from four different action classes. Variability of feature vector patterns expresses the ability of our method to discriminate between various categories. Subsequently, we employ EMK over the output of our 3D kernel descriptor to abstract mid-level features for classification.

Similar to the concept of kernel descriptors, EMK [2] is the kernel representation of well-known bag-of-words method which has been shown to produce more accurate quantization and a better performance as a result.

One way to use match kernels as an alternative to bag-of-words (BOW) method is to add local kernels over all possible combinations of features in two samples (the sum kernels in [31]). This method and lots of other approaches to employ kernels in this manner suffer from space and time complexity as they need to evaluate the full kernel matrix. In contrast, such a computationally expensive operation is not required in EMK. This makes its complexity linear in both time and space. In EMK, local features are projected to a low-dimensional subspace. We then obtain set-level features by computing the mean of the projected feature vectors.

BOW model quantizes local features into a \( D \)-dimensional binary vector \( \mu(x) = [\mu_1(x), \ldots, \mu_D(x)]^T \). Each element \( \mu_i(x) \) is a binary variable, where \( \mu_i(x) = 1 \) indicates \( x \) is in the cluster \( R(v_i) \) and 0 otherwise. Here, \( R(v_i) \) is a cluster of samples with centroid of \( v_i \). We normalize the histogram vector by \( \mu(X) = \frac{1}{|X|} \sum_{x \in X} \mu(x) \). In this work, we use
BOW features together with a nonlinear classifier to achieve high recognition performance. The resulting nonlinear kernel function using a kernel classifier is:

\[ K_B(X, Y) = \bar{\mu}(X)\bar{\mu}^\top(Y) = \frac{1}{|X||Y|} \sum_{x \in X} \sum_{y \in Y} \mu(x)\mu(y) \]

\[ = \frac{1}{|X||Y|} \sum_{x \in X} \sum_{y \in Y} k(x, y) \quad (4) \]

where \( k(x, y) \) is a kernel function used to evaluate the similarity between local features \( x \) and \( y \).

IV. EXPERIMENTS

We test our approach on three standard RGB-D activity datasets including MSR Action 3D [5] dataset, MSR Action Pairs dataset [4], and MSR Gesture 3D dataset [6].

A. Datasets

MSR Action 3D dataset [5] contains 567 depth sequences with resolution of \( 320 \times 240 \). There are 10 subjects each performing an action for two or three times. The dataset consists of 20 action categories: bend, draw x, draw tick, draw circle, forward kick, forward punch, golf swing, hammer, hand catch, hand clap, high arm wave, high throw, horizontal arm wave, jogging, pick up, sideboxing, side kick, tennis swing, tennis serve, throw, and two hand wave.

MSR Action Pairs dataset [4] contains both RGB and depth sequences. This dataset consists of activities with similar motion and shape cues, which makes it challenging for motion cue based methods such as [3]. For example, actions "put down a box" and "pick up a box" have different temporal correlations but are similar in terms of motion.

Using temporal relations of frames in these sequences is the key for accurate classification. There are 12 different actions (6 pairs): "lift a box/place a box", "pick up a box/put down a box", "push a chair/pull a chair", "put on a backpack/take off a backpack", "stick a poster/remove a poster", and "wear a hat/take off a hat". Each action is repeated three times by ten actors, resulting in 360 instances in total.

MSR Gesture 3D dataset [6] contains depth sequences of a group of American Sign Language (ASL) gestures. There are 12 types of gestures in this dataset, which are: bathroom, blue, finish, green, hungry, milk, past, pig, store, where, j, z. Each gesture is performed two or three times by 10 different subjects with the hand portion is captured as the final instance in dataset. The dataset contains 336 depth sequences. For recognition on this dataset, both shape and movement of hands are essential. Self-occlusion is one of the main factors, which makes the dataset a challenging benchmark for action recognition.

B. Experimental Settings

Presenting undefined depth points in depth images as black (zero intensity in gray-scale representation) dots make popular interest point detectors such as STIP [32] to perform poorly in detecting discriminative patches in depth sequences. For gathering maximum amount of information and dealing with the aforementioned problem, we employ dense sampling over 3D patches throughout the whole video. To handle the computational cost of dense sampling, we resize instances to be no larger than \( 150 \times 150 \) in spatial dimensions with preserved ratio. We exploit the efficient match kernels (EMK) [2] and specifically constrained kernel singular value decomposition (CKSVD) for producing the video-level features in our next hierarchical structure. Finally, we use the linear SVM over video-level features for the classification task.

Choosing dataset dependent hyperparameters and running empirical tests to get the best parameters can be an effective way for boosting performance. However, we set some of the parameters fixed during all experiments and try to run practical experiments with parameters of EMK which have a great impact on final results. Kernel parameters in orientation and position kernel are \( \gamma_o = 5 \) and \( \gamma_p = 3 \). The \( \epsilon \) value in computing the normalized gradient magnitude is set to 0.8. We run experiments with different values for the number of basis vectors in CKSVD and its kernel value. In using dense sampling, the overall performance is expected to be better with using multiple patch sizes. However, we empirically choose patches with size \( 16 \times 16 \times 16 \) with 50\% overlap with neighbor patches in sampling for all experiments. Overall accuracy on all the three datasets with various parameters are shown in Figure 5, Table I, and Table II.
C. Results on MSR Action 3D Dataset

We employ our method on this dataset with standard experiment setup [3], where instances from subjects of 1, 3, 5, 7, and 9 are treated as training data. The rest are treated as testing data. Fig. 3 presents the confusion matrix. Our method achieves 92.73% accuracy.

Table III shows the accuracy comparison of different methods. Note that the results from [4] is not as the same as published paper because their experiment is not performed with the exact same setup as other publications. We acquire their code and run the test with standard setup on dividing test and train data which leads to an accuracy of 88.36%. However, for showing our method is general in terms of choosing testing and training data and is not over fitted on standard experimental setup, we run it on all possible permutations of having five subjects for training data and the remaining for testing data. Result is 252 runs for choosing five subjects out of ten. We run the experiments with the same set of hyperparameters, and obtain an accuracy of 82.40 ± 3.63% (mean ± std). This shows that our method is not dependent on a specific permutation for choosing training and testing data.

D. Results on MSR Action Pairs Dataset

We follow the same training/testing protocol as in [4], where samples of the first five subjects are utilized for training and the rest are for testing. We illustrate the confusion matrix in Fig. 3(b). Our method achieves 98.33% accuracy. Actions “stick a poster” is confused with “remove a poster”, and “place a box” is confused with “lift a box”, due to visual similarity.

Table IV shows the accuracy comparison between our method and previous work [35], [3], [4] as reported in [4]. Our method outperforms comparison methods on MSR Action Pairs dataset. Capturing the temporal changes in our 3D kernel descriptor gives this method the ability to differentiate between 3D patches with the same shape and different motion direction.

E. Results on MSR Gesture 3D Dataset

We test our method using cross-subject test where in each test we use the data gathered from one person for testing and the remaining nine persons for training. The confusion matrix of our approach is given in Fig. 3(c). We achieve a recognition accuracy of 95.66%. Confusions occur between “past” and “hungry”, “finish” and “milk”, “j” and “blue”, and “green” and “store”. Examples of misclassifications are illustrated in Fig. 4.
Our method is also compared with existing methods [4], [16], [36], [6]. Table V addresses our performance on this comparison dataset compared to other comparison methods. Results demonstrate that our method significantly outperforms these comparison methods. This is caused by the hierarchical kernels developed in this paper. The proposed 3D gradient kernel can better summarize local motion information in 3D patch. In addition, the EMK kernel used in our method is capable of learning nonlinear correlations of human body parts, and provide rich mid-level features for classification.

V. CONCLUSION

We have proposed a simple and effective method for employing kernel descriptors in describing the 3D geometric surface of depth sequences (oriented normal vectors) [7] in human action recognition task. Our descriptor is an extension of gradient kernel descriptor in [1] which is shown to be an effective way to capture pixel-level attributes in object recognition. In next level we use EMK to abstract mid-level features to produce video-level representations. Through extensive experiments, we show that our method can achieve better performance over state-of-the-art methods on standard RGB-D action datasets.

Though effective, our method is computational expensive due to dense sampling. Efficient sampling scheme will be explored future work.

REFERENCES