Recognizing Human Actions Using Key Poses

Sermetcan Baysal, Mehmet Can Kurt and Pınar Duygulu
Bilkent University, Department of Computer Engineering, 06800, Ankara, Turkey
{sermetcan, kurt, duygulu}@cs.bilkent.edu.tr

Abstract

In this paper, we explore the idea of using only human pose, without utilizing any temporal information, for action recognition. In contrast to the other studies using complex action representations, we propose a simple method, which relies on extracting "key poses" from action sequences. Our contribution is two-fold. Firstly, representing the pose in a frame as a collection of line-pairs, we propose a matching scheme between two frames to compute their similarity. Secondly, after grouping the frames by k-medoids clustering to extract candidate key poses, we rank the potentiality of each candidate becoming a key pose, by means of a learning algorithm. Our experimental results on KTH dataset have shown that pose information by itself is quite effective in grasping the nature of an action and sufficient to distinguish one from the others.

1. Introduction

Recognizing human actions has become a popular research topic of computer vision. A reliable and effective solution to this problem is essential for a large variety of applications ranging from video surveillance and monitoring to human computer interaction systems.

There are different ways to represent actions and extract features for action recognition. In some studies motion-based methods [3, 4, 17] are exploited, whereas actions can also be defined as space-time shapes [1, 8] or space-time interest points [2, 12, 13, 15] for feature extraction. Moreover, in [9] shape and motion based prototype trees were constructed and in [14] form and motion features were combined for action recognition.

In contrast to the complex representation of actions in the methods above, given the available actions, the human brain can more or less recognize what a person is doing even by looking at a single frame without examining the whole sequence. Therefore, we claim that the pose of the human body contains significant information that can be utilized for action recognition. In this paper, we explore the idea of using only the pose information for action recognition.

Recently, pose information is used in some studies for recognizing action. Ikizler et al. [7] proposes a "bag of rectangles" method that represents the human body as a collection of oriented rectangle patches and uses spatial oriented histograms. Thurau et al. [16] extends Histogram of Oriented Gradients (HOG) based descriptor to represent pose primitives. In [6], Ikizler et al. defines a new shape descriptor based on the distribution of lines fitted to boundaries of human figures and uses line histograms. All of these studies share a common property of employing histograms to represent the pose information present in each frame. However, using histograms for pose representation results in the loss of spatial information among the components (e.g. lines or rectangles) forming the pose. Although dividing the body into a grid structure and combining partial histograms are proposed, this is still not a complete solution. For action recognition, such a loss is intolerable since the configuration of the components is very crucial in describing the nature of a human action involving limb and joint movements. At this point, our work differs from the previous studies by preserving and utilizing spatial information encapsulated in poses.

In this paper, we present a simple method to recognize actions using "key poses". We define key poses as a set of frames that uniquely distinguishes an action from others. We represent the pose in a frame as a collection of line-pairs. For each action, we extract a set of key poses. Given an action sequence, each frame is individually labeled as one of the available actions by comparing it with the key poses. Finally, the action sequence is classified using majority voting. In the following sections, details of each step will be explained.

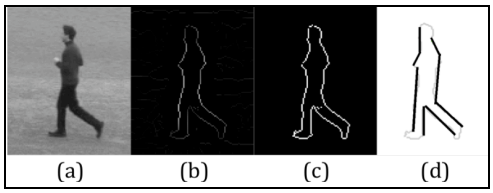

Figure 1. Steps of pose extraction

2. Pose Extraction

Steps of pose extraction can be seen in Figure 1. First, by running a basic correlation-based tracking algorithm we spot the human figure in each frame and crop it by a bounding box (a). Then, we compute the global probability of boundaries (GPB) [11] to extract the edge information (b). To eliminate the effect of noise caused by short and/or weak edges in cluttered backgrounds, we next apply hysteresis thresholding (c). We find the optimal low and high threshold values for a given frame sequence as follows: first, one random frame is selected from each action sequence, and then the edges of the human figure are marked manually by using a polygon. The deviation of the edge probability values lying in the selected region is utilized to assign low and high threshold. To eliminate the remaining noise further, we project the edgels (edge pixels) on x and y-axis, then remove the pixels that do not belong to the largest connected component. Then, edgels (c) are chained by using closeness and orientation information. The edgel-chains are partitioned into roughly straight contour segments. This chained structure is used to construct a contour segment network (CSN) as seen in (d). Finally, we use k-Adjacent Segments (k-AS) descriptor, introduced by Ferrari et al. in [5], which is becoming popular in object recognition area.

A group of k segments is a k-AS if and only if the i^{th} segment is connected in the CSN to the $(i + 1)^{th}$ one, for $i \in \{1 \ldots k-1\}$. Human pose, especially limb movements, can be better represented by using L-shapes. Therefore, in our work we select k = 2, and refer to 2-AS features as line-pairs.

Each line-pair consisting of lines s_1 and s_2 is represented with the following vector:

$$V_{line-pair} = \left(\frac{r_2^x}{N_d}, \frac{r_2^y}{N_d}, \theta_1, \theta_2, \frac{l_1}{N_d}, \frac{l_2}{N_d} \right) \quad (1)$$

where $r_2 = (r_2^x, r_2^y)$ is the vector going from midpoint of s_1 to midpoint of s_2, θ_i is the orientation and $l_i = ||s_i||$ is the length of s_i (i = 1,2). N_d is the distance between the two midpoints, which is used as the normalization factor.

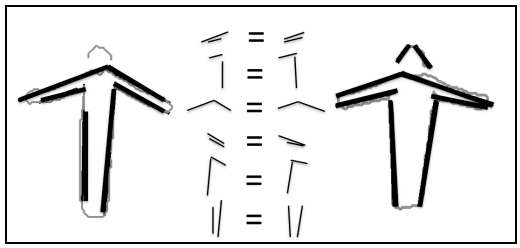

Figure 2. Matched line pairs in similar poses

3. Calculation of Similarity Between Poses

Each frame in a given action sequence is represented by a line-pair descriptor consisting of vectors as shown in equation 1. The similarity between two line-pair vectors v_a and v_b is given by the following formula as suggested in [5]:

$$d(a,b) = w_r \cdot \left\| r_2^a - r_2^b \right\| + w_\theta \cdot \sum_{i=1}^{2} D_\theta\left(\theta_i^a, \theta_i^b\right) + \sum_{i=1}^{2} \left| \log\left(l_i^a / l_i^b\right) \right| \quad (2)$$

where the first term is the difference in the relative locations of the line-pairs, the second term measures the orientation difference of the line pairs and the last term accounts for the difference in lengths. The weights of the terms are $w_r = 4$ and $w_\theta = 2$. Note that Eq. 2 does not compute the overall similarity between two frames consisting of multiple line-pairs.

In our study, to calculate the similarity between two frames, we compare their line-pair descriptors. Any two line-pair descriptors can mathematically be thought of as two sets with different cardinalities. In order to match elements of these two sets, we adopt the characteristics of bijective functions of mathematics in which there is a one-to-one correspondence between those sets; i.e. both 'one-to-one' and 'onto'.

Let f_1 and f_2 be two frames having line-pairs descriptors $\Phi_1 = \{v_1^1 \ldots v_i^1 \ldots v_n^1\}$ and $\Phi_2 = \{v_1^2 \ldots v_k^2 \ldots v_m^2\}$ with number of line-pair vectors n and m, respectively. We compare each line-pair vector v_i^1 in Φ_1 with each line-pair vector v_k^2 in Φ_2 to find matching line-pairs. v_i^1 and v_k^2 are matching line-pairs if and only if among vectors in Φ_2, v_k^2 has the minimum distance to v_i^1 and among vectors in Φ_1, v_i^1 has the minimum distance to v_k^2. With this constraint the 'one-to-one' matching property is satisfied. In Figure 2, we illustrate matching line-pairs between two similar poses. We take the average of the matched line-pair distances and denote it by d_{avg}. Finally, if the 'onto' property is not satisfied, we penalize d_{avg} value with;

$$penalty = \frac{\min(m,n)}{\left| match(f_1, f_2) \right|} \quad (3)$$

where $|match(f_1, f_2)|$ denotes the number of matched line-pairs between f_1 and f_2. Finally, similarity between f_1 and f_2 is computed as;

$$sim(f_1, f_2) = d_{avg} \cdot penalty^p \quad (4)$$

We empirically found that the optimal value for p is 2.

4. Finding Key Poses

Intuitively, to find key poses, it is reasonable to group the frames, which show common pose appearances. Thus, our key pose extraction process bases on k-medoids clustering algorithm since the cluster medoids tend to represent common poses in each action and they are potential candidates for key poses. However, using medoids directly as key poses does not guarantee that they distinguish an action from others since some set of poses may belong to multiple actions. For example, handclapping and handwaving actions of the KTH dataset [15] share instants where the human figure is facing the camera with arms sticking to the body. Therefore, we apply a post-processing step in which, by a learning algorithm, we rank the potentiality of each candidate distinguishing an action from others and becoming a key pose. Finally, we sort the candidate key frames for each action according to their potentiality scores and select top-K highest ranked frames as key poses. (The highest ranked key poses for different actions can be seen in Figure 3.)

Algorithm 1. Finding key poses

1. For K = 1 to N
- *1.1. For each action $a_i \in A$, where $A = \{a_1 \dots a_M\}$*
  - *1.1.1. Cluster all training frames belonging to a_i by running K-medoids algorithm and obtain K clusters.*
  - *1.1.2. Take cluster medoids as a set of candidate key poses c_i for action a_i, where $c_i = \{c_{i1} \dots c_{iK}\}$*
- *1.2. For each frame f in the set*
  - *1.2.1. Compare f with the key pose set $\{c_1 \dots c_M\}$*
  - *1.2.2. Let c_{xy} be the nearest neighbor of f, where $x \in [1,M]$ and $y \in [1,K]$*
  - *1.2.3. If label(c_{xy}) = label(f) then increment score(c_{xy})*
  - *1.2.4. Else decrement score(c_{xy})*

2. Sort score values to obtain the ranked list

Figure 3. Key poses for 6 different actions (boxing, hand clapping, hand waving, jogging, running, walking) of the KTH dataset [15]

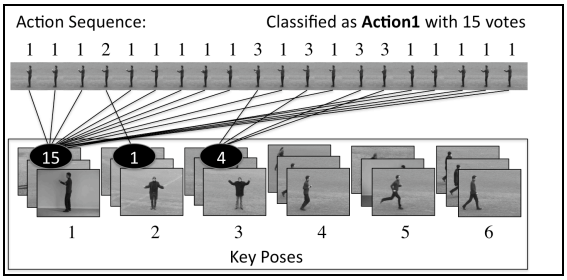

Figure 4. Action recognition using key poses

5. Recognizing Actions

In order to classify a given action sequence, each frame is compared with K number of key poses of each action and assigned the action label of the most similar key pose. Intuitively, for instance, a boxing sequence is expected to consist of frames that are more similar to boxing key poses. Therefore, we use majority voting to classify the sequence (Figure 4).

6. Experimental Results

We tested our action recognition algorithm on the KTH dataset [15], which includes 6 actions (boxing, handclapping, handwaving, jogging, running, walking) performed by 25 different actors in 4 scenarios; outdoors (s1), outdoors with scale variation (s2), outdoors with different clothes (s3), indoors (s4). In our experiments, we regarded the dataset as a single large set (all-scenarios-in-one) with the exception of some samples having extensive noise in their edge detection results.

To evaluate our classification performance, we applied 10-fold cross-validation and averaged the results. On each run, we used 75% of the set for training and the remaining 25% for testing. These sets were selected randomly. In the KTH dataset, actions are performed with varying periodicity. For consistency, as in [14], we trim action sequences to 20-50 frames so that the action is performed only once.

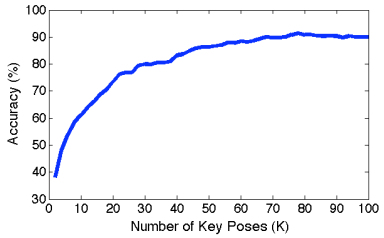

Figure 5. Classification accuracy vs. number of poses per action (K) graph for KTH dataset

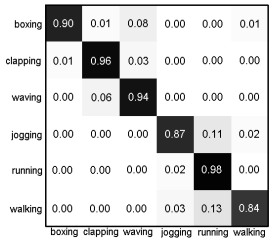

Figure 6. Confusion matrix on KTH at K=78

Figure 5 illustrates the variation of average classification accuracy with respect to the number of key poses per action (*K*). Classification accuracy rises as we increase *K* up to some point because, distinct actors may perform an action in different ways. We obtained a recognition rate of 91.5% at *K*=78. The all-scenarios-in-one recognition results of different methods on the KTH dataset vary between 71.72% [15] and 93.80% [10]. Our method provides better results than the majority of the related studies; only a few (best ones) show higher performance with a minor difference. If we look at our misclassifications in Figure 6, we can see that mainly 'jogging' and 'walking' are confused with 'running', which is reasonable considering their visual similarity. More than half of our misclassifications belong to samples from scenario s3, because actors carry bags and wear different clothes, which has a negative effect on pose extraction.

7. Summary and Discussion

In this paper, we propose a method for human action recognition by exploiting the pose information in a given action sequence. We embody the shape features present in each frame as line-pairs and create a descriptor, which stores the position, orientation and length information. Therefore, in contrast to the other studies in the literature, which encode pose information with histograms, our approach is able to preserve the spatial relations of the components forming the boundaries of a human figure. By means of a powerful matching mechanism, we extract the key frames, with a learning algorithm. Since it relies on good edge detection, the sensitivity to the noise in cluttered backgrounds appears as the biggest downside of our approach.

Providing better results than the most of the related studies on KTH dataset, we show that pose information by itself is quite effective in grasping the nature of an action and sufficient to distinguish one from the others. It is apparent that the overall recognition performance can be enhanced by including the local and/or global motion information.

References

[1] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. *Actions as Space-Time Shapes*. ICCV, 2005.

[2] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. *Behavior Recognition via Sparse Spatio-Temporal Features*. VS-PETS, 2005.

[3] A. A. Efros, A. C. Berg, G. Mori, and J. Malik. *Recognizing Action at a Distance*. ICCV, 2003.

[4] A. Fathi and G. Mori. *Action Recognition by Learning Mid-Level Motion Features*. CVPR, 2008.

[5] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid. Groups of adjacent contour segments for object detection. *IEEE Trans. Pattern Anal. Mach. Intel l.*, 30(1):36–51, 2008.

[6] N. Ikizler, R. G. Cinbis and P. Duygulu, *Human Action Recognition With Line and Flow Histograms*, ICPR, 2008.

[7] N. Ikizler and P. Duygulu, Histogram of Oriented Rectangles: A New Pose Descriptor of Human Action Recognition. *Image and Vision Computing*, 27(10):1515-1526, 2009.

[8] Y. Ke, R. Sukthankar, and M. Hebert. *Spatio-Temporal Shape and Flow Correlation for Action Recognition*. In Visual Surveillance Workshop, 2007.

[9] Z. Lin, Z. Jiang, and L. S. Davis. *Recognizing Actions by Shape-Motion Prototype Trees*. ICCV, 2009.

[10] J. Liu, J. Luo, and M. Shah. *Recognizing Realistic Actions from Videos in the Wild*. CVPR, 2009.

[11] M. Maire, P. Arbelaez, C. Fowlkes, J. Malik. *Using Contours to Detect and Localize Junctions in Natural Images*. CVPR, 2008.

[12] J. C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised Learning of Human Action Categories Using Spatial-Temporal Words. *Int'l J. Computer Vision*, 79(3):299–318, 2008.

[13] S. Nowozin, G. Bakir, and K. Tsuda. *Discriminative Subsequence Mining for Action Classification*. ICCV, 2007.

[14] K. Schindler and L. V. Gool. *Action Snippets: How Many Frames Does Human Action Recognition Require*? CVPR, 2008.

[15] C. Schuldt, I. Laptev, and B. Caputo. *Recognizing Human Actions: A Local SVM Approach*. ICPR, 2004.

[16] C. Thurau and V. Hlavac. *Pose Primitive Based Human Action Recognition in Videos or Still Images*. CVPR, 2008.

[17] Y. Wang, P. Sabzmeydani, and G. Mori. *Semi-Latent Dirichlet* Allocation:*A Hierarchical Model for Human Action Recognition*. ICCV Workshop on Human Motion, 2007