Recognizing Human Actions Using Key Poses

Sermetcan Baysal, Mehmet Can Kurt and Pinar 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 pose, without utilizing any temporal information, for human 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, to extract "key poses" for each action, we present an algorithm, which selects the most representative and discriminative poses from a set of candidates. Our experimental results on KTH and Weizmann datasets have shown that pose information by itself is quite effective in grasping the nature of an action and sufficient to distinguish one from others.**

I. 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 in others actions are 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. Being motivated by this observation, in this study, we ignore any temporal information and explore the potentiality of using only pose information in recognizing human actions.

Recently, pose information is used in some studies for recognizing actions. Ikizler et al. [7] propose 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] extend Histogram of Oriented Gradients (HOG) based descriptor to represent pose primitives. In [6], Ikizler et al. define a new shape descriptor based on the distribution of lines fitted to boundaries of human figures and use 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 –even if grid structures are used for localization– results in the loss of spatial information among the components (e.g. lines or rectangles) forming the pose. 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. More importantly, temporal information is totally disregarded.

In this paper, we present a simple method to recognize actions using "key poses", which are defined 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, a set of key poses is extracted. 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, each step will be explained in detail.

II. POSE EXTRACTION

Poses in each frame are extracted following the steps shown in Fig. 1. First, by running a basic correlation-based tracking algorithm, human figure in each frame is spotted and cropped to form a bounding box (a). Next, the global probability of boundaries (GPB) [11] is computed to extract the edge information (b). To eliminate the effect of noise caused by short and/or weak edges in cluttered backgrounds, hysteresis thresholding is applied as the next step. At this point, the optimal low and high threshold values are found for a given frame sequence as follows: first, one random frame is selected from each action sequence, then the edges of the human figure are marked manually by using a polygon. The variation of the edge probability values lying in the selected region is utilized to assign low and high threshold. To eliminate the remaining noise further, the edgels (edge pixels) are projected on x and y-axis, then the pixels that do not belong to the largest connected component are removed. Afterwards, the remaining 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, the CSN is represented by k-Adjacent Segments (k-AS) descriptor, introduced by Ferrari et al. in [5], which is becoming popular in object recognition area.

1051-4651/10 $26.00 © 2010 IEEE
DOI 10.1109/ICPR.2010.427

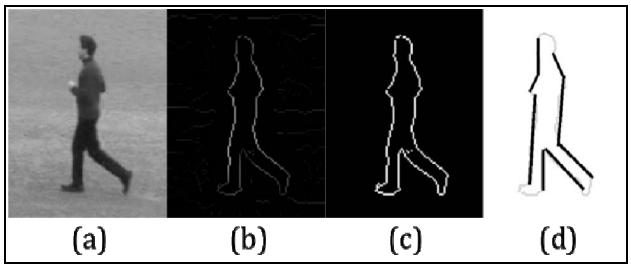

Figure 1. Steps of pose extraction

As defined in [5], 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 $i \in \{1 \ldots k-1\}$. Note that two segments are considered as connected, when they are adjacent along some object contour even if there is a small gap separating them physically. 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 descriptor:

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

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.

III. Calculation of Similarity Between Poses

Each frame in a given action sequence is represented by a set of line-pair descriptors. The similarity between two line-pair descriptors v_a and v_b is computed by the following formula as suggested in [5]:

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

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 computes the similarity only between two individual line-pairs. Therefore, in this study, we introduce a method to compare two frames consisting of multiple line-pairs.

Any two frames consisting of line-pair descriptors can mathematically be thought of as two sets X and Y with different cardinalities. In order to match elements of these two sets, we require both 'one-to-one' and 'onto' properties to be satisfied so that each element in X is associated with exactly one element in Y.

Let f_1 and f_2 be two frames having set of line-pair 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 descriptors n and m, respectively. We

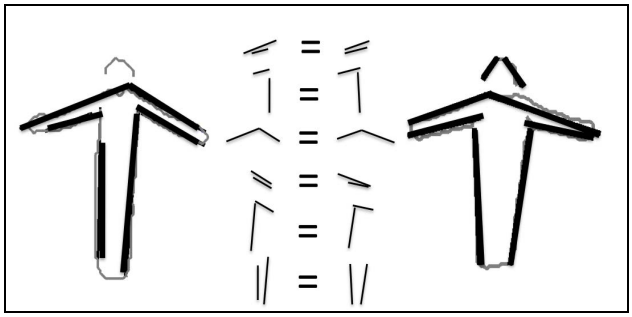

Figure 2. Matched line-pairs in similar poses

compare each line-pair descriptor v_i^1 in Φ_1 with each line-pair descriptor 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 descriptors in Φ_2, v_k^2 has the minimum distance to v_i^1 and among descriptors 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. Fig. 2 illustrates 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:

$$penalty = \min(m,n) \Big/ \left| match(f_1, f_2) \right| \quad (3)$$

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.

IV. Finding Key Poses

Key poses can be described as the ones, which are representatives in a specific action. Intuitively, to find key poses, it is reasonable to group the frames, which show common pose appearances. Thus, we base our key pose extraction process on k-medoids clustering algorithm since the cluster medoids tend to represent common poses in each action. 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 propose a method described in Algorithm 1 to rank the potentiality of each candidate key pose in distinguishing an action from others. 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 6 different actions included in the KTH dataset can be seen in Fig. 3.

Figure 3. Key poses found for 6 different actions (boxing, handclapping, handwaving, jogging, running, walking) of the KTH dataset

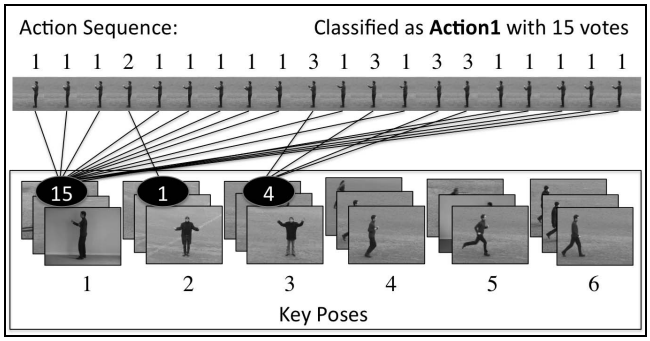

Figure 4. Action recognition using key poses

Algorithm 1. Finding Key Poses

1. For K = 1 to N, where K is the number of clusters
  1.1. For each action $a_i \in A$, where $A = \{a_1 \dots a_M\}$ and M is the number of unique actions
    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 sets $\{c_1 \dots c_M\}$
    1.2.2. Let c_{ik} be the nearest neighbor of f, where $i \in [1,M]$ and $k \in [1,K]$
    1.2.3. If label(c_{ik}) = label(f) then increment score(c_{ik})
    1.2.4. Else decrement score(c_{ik})
2. Sort score values to obtain a ranked list for each action

V. Recognizing Actions

In order to classify a given action sequence, first, each frame is compared to all key poses of all actions and assigned the action label of the most similar key pose. Then, we apply majority voting among these assigned labels. Fig. 4 illustrates the classification process with an example.

VI. Experimental Results

We tested our action recognition algorithm on the Weizmann [1] and KTH [15] datasets. Weizmann dataset contains 9 actions (bend, jack, jump-forward-on-two-legs, jump-in-place-on-two-legs, run, gallop-sideways, walk, wave-one-hand, wave-two-hands), which are performed by 9 different actors. KTH dataset contains 6 actions (boxing, hand-clapping, hand-waving, jogging, running, walking) performed by 25 different actors in 4 scenarios; outdoors (s1), outdoors with scale variation (s2), outdoors with different clothes (s3) and indoors (s4).

For Weizmann dataset, we omitted the noise reduction steps requiring some manual effort (explained in section 2) since we used the available silhouettes to extract our line-pair descriptors. To evaluate our classification performance, we applied leave-one-out cross-validation.

We regarded the KTH dataset as a single large set (all-scenarios-in-one) with the exception of some action instances having extensive noise in their edge detection results. In order to evaluate our classification performance, we applied 10-fold cross-validation and averaged the results. Because of the high computational cost, we randomly selected about half of the data set and applied 10 fold cross validation by using 75% of the samples for training and the remaining 25% for testing in each run. 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.

We obtained recognition rates of 92.6% at K=47 and 91.5% at K=78 for Weizmann and KTH datasets respectively, where K is the number of key poses per action. For Weizmann dataset, we can raise our recognition rate up to 95.06% when we apply a geometrical constraint in which we divide each frame into rectangular grid structures and only allow matching between line-pairs within the same grid. However, for the KTH dataset, the human figures having incomplete body parts (e.g. missing legs and heads) due to poor edge detection makes this geometrical constraint inapplicable and leads to lower classification accuracy. Fig. 5 illustrates the variation of average classification accuracy with respect to the number of key poses (*K*) per action for the KTH dataset. Our method requires large *K* values to achieve good classification performance because distinct actors may perform an action in different ways.

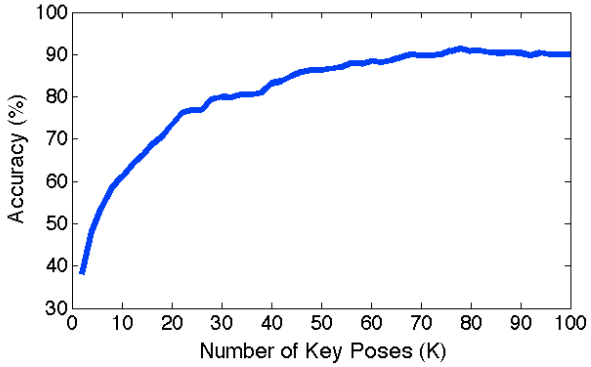

Figure 5. Classification accuracy vs. number of poses per action (K) graph for KTH dataset

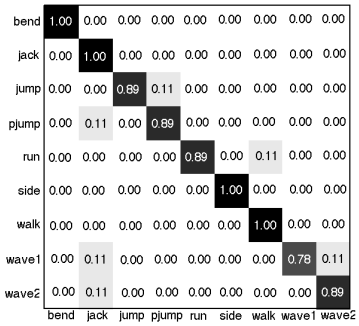

Figure 6. Confusion matrix of the Weizmann dataset at K=47

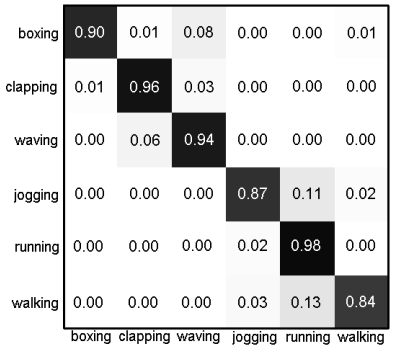

Figure 7. Confusion matrix of the KTH dataset at K=78

The recognition results of different methods in the literature vary between 73% [12] and 100% [7] on the Weizmann dataset and between 71.72% [15] and 93.80% [10] on the KTH dataset. Our results show that the pose information by itself is quite effective in grasping the nature of an action and sufficient to distinguish one from the others.

Fig. 6 and Fig. 7 shows misclassifications of our method. Mainly, visually similar actions (e.g. 'running' and 'walking') are confused with each other. Moreover, most of our misclassifications on the KTH dataset belong to samples from a shooting scenario (s3), where the actors carry bags and wear different clothes, leading to the existence of unexpected line-pairs.

VII. Discussion and Conclusion

In this study, we introduce a new method for representing human pose and explore its ability in recognizing human actions by itself. We embody the shape features present in each frame as line-pairs described by position, orientation and length information. Therefore, in contrast to the other studies in the literature, which encode pose information with histograms, our approach is better in preserving the spatial relations of the components forming the boundaries of a human figure. By means of the proposed matching mechanism, the correspondences between the set of line-pairs in two frames are captured. Then, each action is described by key poses, which are representatives of an action and powerful to distinguish it from the others. Since our method relies on good edge detection, the sensitivity to the noise in cluttered backgrounds appears as the biggest downside of our approach. It is apparent that the overall recognition performance can be enhanced by including the local and/or global motion information and using advanced classification techniques.

Acknowledgment

This project is partially supported by TUBITAK project grant no. 104E065.

References

[1] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. *International Conference on Computer Vision (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. *International Conference on Computer Vision (ICCV)*, 2003.

[4] A. Fathi and G. Mori. Action recognition by learning mid-level motion features. *Conference on Computer Vision and Pattern Recognition (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. *International Conference on Pattern Recognition (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. *Visual Surveillance Workshop*, 2007.

[9] Z. Lin, Z. Jiang, and L. S. Davis. Recognizing actions by shape-motion prototype trees. *International Conference on Computer Vision (ICCV)*, 2009.

[10] J. Liu, J. Luo, and M. Shah. Recognizing realistic actions from videos "in the wild". *Conference on Computer Vision and Pattern Recognition (CVPR),* 2009.

[11] M. Maire, P. Arbelaez, C. Fowlkes, J. Malik. Using contours to detect and localize junctions in natural images. *Conference on Computer Vision and Pattern Recognition (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 classi• cation. *International Conference on Computer Vision (ICCV)*, 2007.

[14] K. Schindler and L. V. Gool. Action snippets: how many frames does human action recognition require? *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[15] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: a local SVM approach. *International Conference on Pattern Recognition (ICPR)*, 2004.

[16] C. Thurau and V. Hlavac. Pose primitive based human action recognition in videos or still images. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[17] Y. Wang, P. Sabzmeydani, and G. Mori. Semi-latent Dirichlet allocation: a hierarchical model for human action recognition. *International Conference on Computer Vision (ICCV), Workshop on Human Motion*, 2007.