Learning self-similarities for action recognition using conditional random fields

Imran N. Junejo
University of Sharjah
U.A.E.

Abstract

Human action recognition is a complex process due to many factors, such as variation in speeds, postures, camera motions etc. Therefore an extensive amount of research is being undertaken to gracefully solve this problem. To this end, in this paper, we introduce the application of *self-similarity surfaces* for human action recognition. These surfaces were introduced by Shechtman & Irani (CVPR'07) in the context of matching similarities between images or videos. These surfaces are obtained by matching a small patch, centered at a pixel, to its larger surroundings, aiming to capture *similarities* of a patch to its neighborhood. Once these surfaces are computed, we propose to transform these surfaces into Histograms of Oriented Gradients (HoG), which are then used to train Conditional Random Fields (CRFs). Our *novelty* lies in recognizing the utility of these self-similarity surfaces for human action recognition. In addition, in contrast to Shechtman & Irani (CVPR'07), we compute only a few of these surfaces (two per frame) for our task. The proposed method does not rely on the structure recovery nor on the correspondence estimation, but makes only mild assumptions about the rough localization of a person in the frame. We demonstrate good results on a publicly available dataset and show that our results are comparable to other well-known works in this area.

1. Introduction

Visual recognition and understanding of human actions has attracted much of the attention over the past three decades Moeslund et al. (2006); Wang et al. (2003); Turaga et al. (2008) and still remains an active research area of computer vision. A good solution to the problem holds a huge potential for many applications such as the search and the structuring of large video archives, video surveillance, human-computer interaction, gesture recognition and video editing. Recent work has demonstrated the difficulty of the problem associated with the large variation of human action data due to the individual variations of people in expression, posture, motion and clothing; perspective effects and camera motions; illumination variations; occlusions and disocclusions; and distracting effects of scenes surroundings. In addition, actions frequently involve and depend on manipulated objects adding another layer of variability.

Various approaches using different constructs have been proposed over the years for action recognition. These approaches can be roughly categorized on the basis of representation used by the researchers. Time evolution of human silhouettes was frequently used as action description. For example, Bobick & Davis (2001) proposed to capture the history of shape changes using temporal templates and Weinland et al. (2006) extend these 2D templates to 3D action templates. Similarly, based on silhouettes, notions of *action cylinders* Syeda-Mahmood et al. (2001), and *space-time shapes*Yilmaz & Shah (2005a); Gorelick et al. (2007) have also been introduced. Recently, researchers have started analyzing video sequences as space-time volumes, built by various *local features*, such as intensities, gradients, optical flow etc Fathi & Mori (2008); Jhuang et al. (2007); Filipovych & Ribeiro (2008). Original work in this area is that of Laptev & Lindeberg (2003), and Niebles et al. (2006); Liu & Shah (2008); Jingen et al. (2008); Mikolajczyk & Uemura (2008); Bregonzio et al. (2009); Rapantzikos et al. (2009) and Gilbert et al. (2008) represent some of the recent work in this area. Using these space-time or other local image features, researchers have also attempted at modeling the complex dynamic human motion by adopting various machine learning approaches Ali et al. (2007); Weinland & Boyer (2008); Jia & Yeung (2008): Hidden Markov Models (HMMs) Brand et al. (1997); Wilson & Bobick (1995); Ikizler & Forsyth (2007); Support Vector Machines (SVMs) Ikizler et al. (2008); Yeffet & Wolf (2009); Prototype Trees Lin et al. (2009); or Conditional Random Fields (CRF) and its variants Sminchisescu et al. (2005); Natarajan & Nevatia (2008), using features such as histograms of combined shape context and edge features, fast-fourier transforms of angular velocities, and blocked-based features of silhouettes etc. Some other works based on multiview geometry or pose learning includes that of Syeda-Mahmood et al. (2001); Yilmaz & Shah (2005b); Carlsson (2003); Rao et al. (2002); Shen & Foroosh (2008); Parameswaran & Chellappa (2006); Ogale et al. (2006); Ahmad & Lee (2006); Li et al. (2007); Lv & Nevatia (2007); Shen & Foroosh (2008), requiring either identification of body parts or the estimation of corresponding points between video sequences.

Our approach builds upon the concept self-similarities as introduced by Shechtman & Irani (2007). For a given action sequence, the approach consists of computing *similarities* of the pose to itself in each frame. This we call as the *self-similarity surface*. This surface has been introduced in context of image and video matching previously by the Shechtman & Irani (2007). They build on the assumption that for a phenomenon or a pattern captured in different forms, even though different representation and their corresponding measures vary significantly, there exists a common underlying visual property of patterns, which is captured in terms of the local intensity properties. However, in their work Shechtman & Irani (2007), these surfaces are computed very densely in an image, whereas we perform very sparse sampling, i.e. we compute only two self-similarity surface for an entire image. Also, in this paper, we introduce the usage of these surfaces for the human action recognition. We believe that this *novel* application of these surfaces is very significant for understanding the human actions, and provides acceptable accuracy compared to other well-known methods.

In the rest of the paper we operationalize self-similarity surface for human action sequences. The rest of the paper is organized as follows. In the next section we review related work. Section 3 gives a formal definition of self-similarity surface using image color features. Section 4 describes the representation and training of action sequences based on HoG descriptors, constructed from the self-similarity surfaces. In Section 5 we test the method on public dataset and demonstrate the practicality and the potential of the proposed method. Section 6 concludes the paper.

2. Related Work

The methods most closely related to our approach are that of Shechtman & Irani (2007); Benabdelkader et al. (2004); Cutler & Davis (2000); Carlsson (2000). Recently for image and video matching, Shechtman & Irani (2007) explored *local* self-similarity descriptors. The descriptors are constructed by correlating the image (or video) patch centered at a pixel to its surrounding area by the sum of squared differences. The correlation surface is transformed into a binned log-polar representation to form a local descriptor used for image and video matching. Differently to this method, we explore the structure of similarities between *all* pairs of time-frames in a sequence. The main focus of our work is on the use of self-similarities for action recognition which was not addressed in Shechtman & Irani (2007). The Figure 3 shows the self-similarity descriptor extracted from two separate images. The figure on the top left has three marked points: 1,2 and 3. The right three images in the first row shows the computed self-similarity descriptor for each of these marked points, respectively. In row two, the leftmost image also has three points marked at almost the same location as the one in row one above. The corresponding self-similarity descriptors are shown in the second row as well. This image demonstrates that even when we have difference images containing same phenomenon, (even in the presence of some perspective distortion), the computed self-similarity descriptors, as can be seen above, bear great similarities. Consequently, the Figure 2 shows the self-similarity descriptors at work. Figure 2(a) shows an input image. A self-similarity descriptor of this image is extracted which is then matched to the descriptors extracted from a database of a large number of images. In the figure, red corresponds to the highest similarity values.

Our approach has a closer relation to the notion of video self-similarity used by Benabdelkader et al. (2004); Cutler & Davis (2000). In the domain of periodic motion detection, Cutler and Davis Cutler & Davis (2000) track moving objects and extract silhouettes (or their bounding boxes). This is followed by building a 2D matrix for the given video sequence, where each entry of the matrix contains the absolute correlation score between the two frames *i* and *j*. Their observation is that for a periodic motion, this similarity matrix will also be periodic. To detect and characterize the periodic motion, they use the Time-Frequency analysis. Following this, Benabdelkader et al. (2004) use the same construct of the self-similarity matrix for gait recognition in videos of walking people. The periodicity of the gait creates diagonals in the matrix and the temporal symmetry of the gait cycles are represented by the cross-diagonals. In order to compare sequences of different length, the self-similarity matrix is subdivided into small units. Both of these works focus primarily on videos of walking people for periodic motion detection and gait analysis. The method in Carlsson (2000) also concerns gait recognition using temporal similarities between frames of different image sequences. None of the methods above explores the notion of self-similarity for action recognition. In addition, we perform very sparse sampling of the foreground space.s

2.1 Overview of our approach

The overview of the proposed approach is as shown in Fig. 1. Whereas, Shechtman & Irani (2007) compute the self-similarity based descriptor densely, we divide the foreground into just two portions, the top and the bottom, as shown in the figure. What we do is basically match the center of the top patch with its surroundings, within a certain radius. And we repeat the same process for the bottom part of the foreground. This results in two self-similarity surfaces, explained below, which are then converted into HoG based descriptors Dalal & Triggs (2005). Once we have these pose descriptors for all action sequences of all classes, we train a Condi-

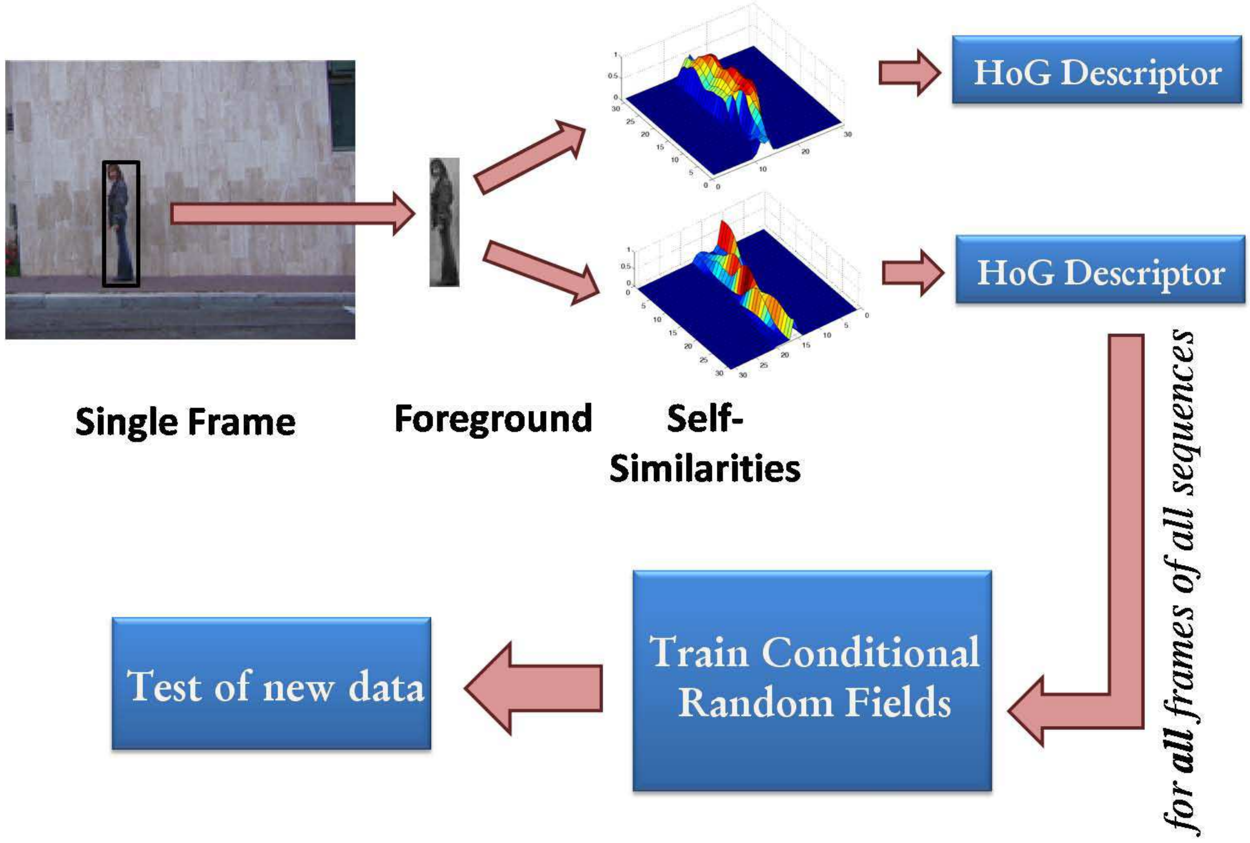


Fig. 1. Overview of the proposed solution. Initially, the foreground is extracted from an action sequence. Once extracted, the foreground is divided into the upper part and the lower part. Each part is used to compute the self-similarity descriptor, as shown in right above. Each surface is then transformed into HoG features. These features are computed for all sequences of each action class and a probabilistic model i.e. Conditional Random Fields, is learned for performing accurate action recognition.

tion Random Field Lafferty et al. (2001), during our training phase. During the testing phase, a test sequence also goes through the same process, and is assigned a probable sequence label that maximizes our conditional model.

3. Self-Similarity Surfaces

In this section we define self-similarities computed from an action sequence. The main contribution of the paper is the introduction of this self-similarity descriptor for action recognition, with the rationale that poses from different action sequences produce self-similarities of distinct patterns or structures, thus allowing us to perform action recognition.
Originally introduced by Shechtman & Irani Shechtman & Irani (2007), the descriptor captures internal geometric layout of local self-similarities within images by using only the color information. Essentially, what it does is capture self-similarity of edges, color, repetitive patterns and complex textures in a simple and unified way Shechtman & Irani (2007). The notion of

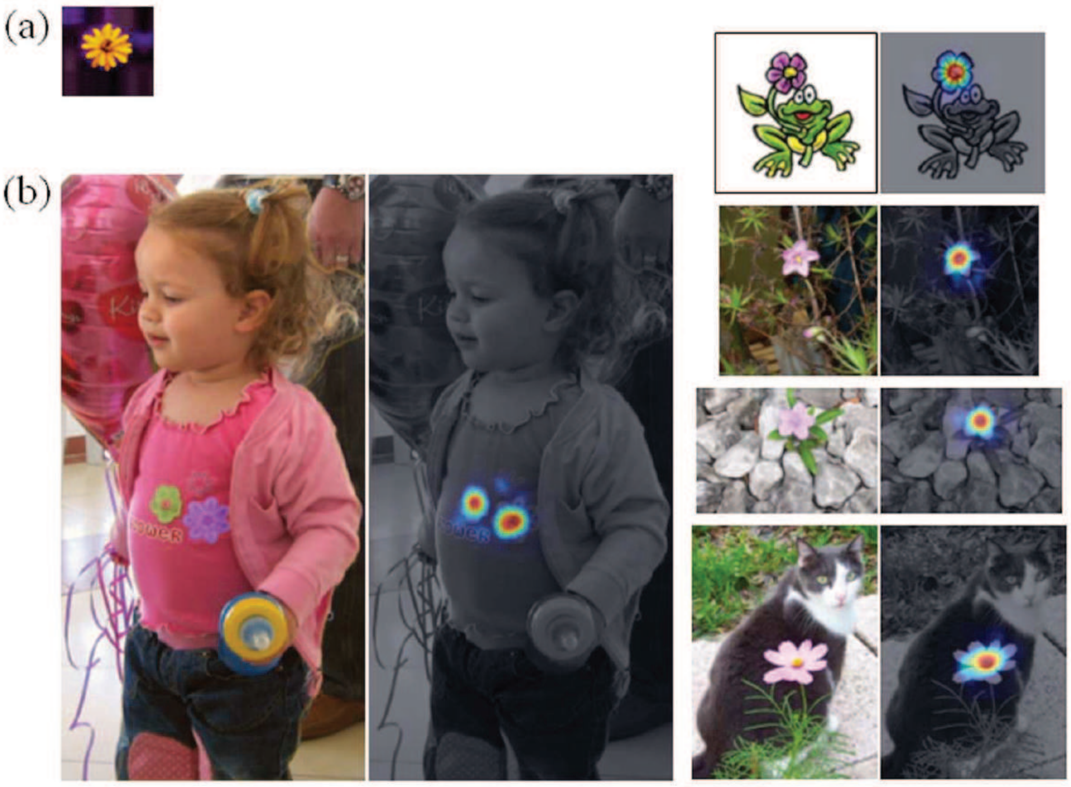


Fig. 2. The figure above shows the matching capabilities of the self-similarity descriptors. (a) shows the input (or the test) image. A self-similarity descriptor of this image is extracted. Once this is done, the descriptor is efficiently matched to the descriptors extracted from a database of a large number of images. In the figure above, red corresponds to the highest similarity values. (image courtesy of Shechtman & Irani (2007)

self-similarity is closely related to the notion of statistical co-occurrence, which is captured by the Mutual Information (MI). An example of this is shown in Figure 1.

The self-similarity descriptor is computed as follows: First, the object (or the actor) is extracted from the action sequence. This can be done by simple application of any background subtraction method (we use the extracted foregrounds provided by Shechtman & Irani (2007)). Once such a foreground is obtained, we divide it into two equal parts (the upper and the lower part). The center p of each patch, generally represented by a 5×5 patch, is compared to the surrounding patches within a radius (generally of size 15 or 30, depending on the size of the foreground object). The comparison of the patches is made by a simple application of *sum of square differences* (SSD). The result surface $\Upsilon_p(x,y)$, is then normalized into a correlation surface:

$$S_p = \exp\left(-\frac{\Upsilon_p(x,y)}{\sigma_{auto}}\right) \tag{1}$$

where σ_{auto} is a constant that takes into account noise, and common variations in color, illumination etc (for our experiments, we set its value to 2.5).

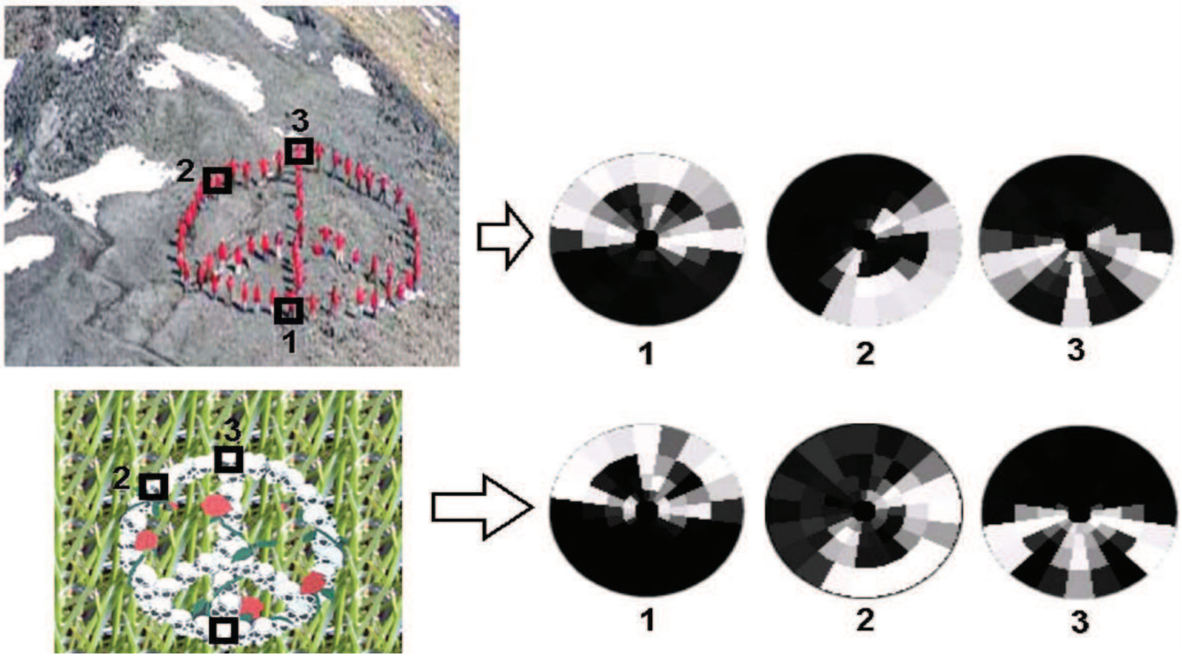


Fig. 3. The figure above shows the self-similarity descriptor extracted from two separate images. The figure on the top left has three marked points: 1,2 and 3. The right three images in the first row shows the computed self-similarity descriptor for each of these marked points, respectively. In row two, the leftmost image also has three points marked at almost the same location as the one in row one above. The corresponding self-similarity descriptors are shown in the second row as well. This image demonstrates that even when we have difference images containing same phenomenon, (even in the presence of some perspective distortion), the computed self-similarity descriptors, as can be seen above, bear great similarities. (image courtesy of Shechtman & Irani (2007)

The surface S_p is then transformed to better distinguish the spatial appearances. To this end, for describing the spatial appearance of a person at each image frame, we compute Histograms of Oriented Gradients (HoG) Dalal & Triggs (2005). This feature, originally used to perform human detection, characterizes the local shape by capturing the gradient structure. In our implementation, we use 8 bin histograms for each of 5×7 blocks defined on (the upper and the lower parts of the) bounding box around the person in each frame. The final self-similarity descriptor $\mathbf{D}^i$, for an image i, is then a concatenation of HoG features obtained from the upper and the lower part of the bounding box.
As shown in the 5, in addition to the SSDs based on the color information, we also test on self-similarity surfaces computed from optical flows. The optical flow is computed by Lucas and Kanade method Lucas & Kanade (1981) on person-centered bounding boxes using two consecutive frames.

4. Modeling & Recognition

At this stage, for each action sequence, we have obtained self-similarity surfaces, two for each frame. As described above, this self-similarity surface is then converted into HoG features. In this section, we aim to learn these features for each action class to perform action recognition. For this purpose, we chose Conditional Random Fields (CRFs) Lafferty et al. (2001)(cf. Fig. 4a). To the best of our knowledge, no work exists that learns these self-similarity surfaces for action recognition.

4.1 Conditional Random Fields

CRFs are a probabilistic framework for segmenting and labeling sequence data. Exhibiting many advantages over the traditional Hidden Markov Models (HMMs), CRFs provide a great flexibility by relaxing the conditional independence assumption generally made for the observation data.

The general framework for CRFs is as follows: Let $\mathbf{X}$ be a random variable over data sequence to be labeled and let $\mathbf{R}$ be a random variable over our corresponding label sequences. All components of $\mathbf{R}_i$ of $\mathbf{R}$ are assume to range over a finite label sequence $\mathcal{R}$, which in our case can be action sequences like `bend, wave, jump, hop, run, walk` etc. Generally, in training dataset, the random variables $\mathbf{R}$ and $\mathbf{X}$ are jointly distributed, but in the case of CRFs we construct the conditional model $p(\mathbf{R}|\mathbf{X})$, rather than explicitly modeling the marginal $p(\mathbf{X})$:

Let $\mathcal{G} = (V, E)$ be an undirected graph over our set of random variables $\mathbf{R}$ and $\mathbf{X}$ (cf. Fig. 4b). Then $(\mathbf{R}, \mathbf{X})$ is a conditional random field in case, when conditioned on X, the random variable $\mathbf{R}_i$ obey the Markov property with respect to the graph: $p(\mathbf{R}_i|\mathbf{R}_j, \mathbf{X}, i \neq j) = p(\mathbf{R}_i|\mathbf{R}_j, \mathbf{X}, i \sim j)$, where $\sim$ means i and j are neighbors in $\mathcal{G}$ Lafferty et al. (2001). Let $\mathcal{C}(\mathbf{X}, \mathbf{R})$ be the set of maximal clique in $\mathcal{G}$, then the CRFs define the conditional probability of the label sequence given the observed sequence as

$$p_\theta(\mathbf{R}|\mathbf{X}) = \frac{1}{Z(\mathbf{X})} \prod_{c \in \mathcal{C}(\mathbf{R},\mathbf{X})} \phi_\theta^c(\mathbf{R}_c, \mathbf{X}_c) \tag{2}$$

where $Z(\mathbf{X})$ is the normalization factor over all states of the sequences, and is given as:

$$Z(\mathbf{X}) = \sum_{\mathbf{R}} \prod_{c \in \mathcal{C}(\mathbf{R},\mathbf{X})} \phi_\theta^c(\mathbf{R}_c, \mathbf{X}_c) \tag{3}$$

and ϕ_θ^c is the potential function of the clique c, and characterizes according to the set of selected features f_θ so that:

$$\phi_\theta^c(\mathbf{R}_c, \mathbf{X}_c) = \exp\left(\sum_{t=1}^{T} \sum_n \lambda_n f_\theta(\mathbf{R}_c, \mathbf{X}_c, t)\right) \tag{4}$$

where the model parameters $\psi = \{\lambda_n\}$ are a set of real weights, per feature. Each feature function $f_\theta(\mathbf{R}_c, \mathbf{X}_c, t)$ is either a state function $s_k(\mathbf{r}_t, \mathbf{x}_t, t)$ or a transition function $g_k(\mathbf{r}_{t-1}, \mathbf{r}_t, \mathbf{x}_t, t)$. State functions depend on a single hidden variable in the model, while the transition function can depend on a pair of hidden variables Lafferty et al. (2001).

Linear-chain CRFs, as shown in Fig. 4b, are widely used in many applications. Accordingly, the cliques of such a conditional model include pair of neighboring sates $(\mathbf{r}_{t-1}, \mathbf{r}_t)$, whereas the connectivity among the observation is unrestricted. Therefore, arbitrary complex observation dependencies can be added to the model without out affected complicating the inferences, as these observations are known and fixed.

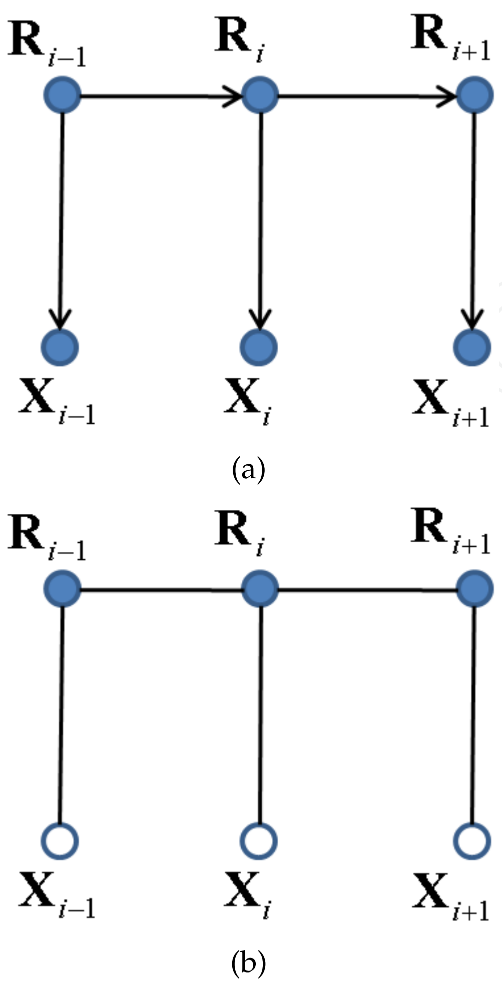


Fig. 4. (a) Graphical structures of simple HMMs. (b) Graphical Structure in the case of CRFs, where the open circle indicates that the data is not generated by the model.

4.2 Parameter Learning

Given a fully labeled training data set $\mathcal{D} = \{\mathbf{R}^i, \mathbf{X}^i\}_{i=1}^N$, the CRF parameters can be obtained by optimizing the conditional log-likelihood function

$$\mathcal{O}(\theta) = \sum_{i=1}^{N} \log p_\theta(\mathbf{R}^i | \mathbf{X}^i) \tag{5}$$

The derivative of (5) with respect to the λ_k associated with clique c is given as:

$$\begin{aligned} \frac{\partial \mathcal{O}(\theta)}{\partial \lambda_k} &= \sum_i \sum_t f_\theta(\mathbf{R}^i_{t,c}, \mathbf{X}^i, t) \\ &\quad - \sum_i \sum_t \sum_{c \in \mathcal{C}} \sum_{\mathbf{R}_c} p_\theta(\mathbf{R}_c | \mathbf{X}^i_t) f_\theta(\mathbf{R}_{t,c}, \mathbf{X}^i, t) \end{aligned} \tag{6}$$

where $\mathbf{R}_{t,c}$ denotes the variable $\mathbf{R}$ at time stamp t in clique c of the CRF, and $\mathbf{R}_c$ ranges over assignment to c.

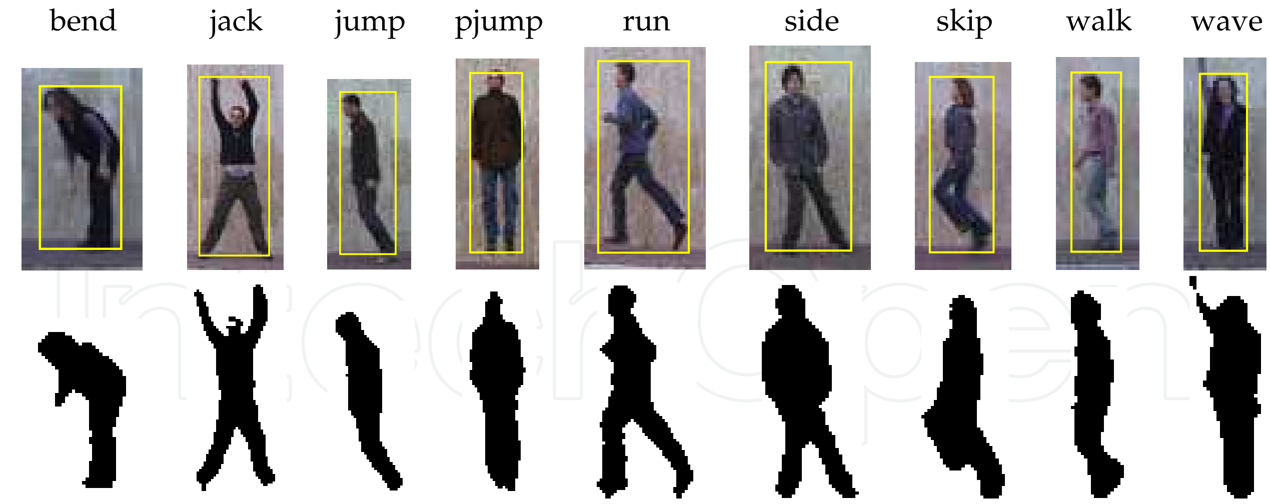


Fig. 5. Weizman dataset: The top row shows instances from the nine different action sequences. The bottom row depicts the extracted silhouettes. The dataset contains ninety-three action sequences videos of varying lengths, each having a resolution of 180 × 144. The nine difference action are: `bending down`, `jumping back`, `jumping`, `jumping in place`, `galloping sideways`, `running`, `walking`, `waving one hand` and `waving two hands`.

To reduce over-fitting, generally a penalized likelihood is used for training the parameters: $\mathcal{O}(\theta) + \frac{1}{2\sigma^2}\|\theta\|^2$ where the second term is the log of a Gaussian prior with variance σ^2, i.e. $P(\theta) = \exp(\frac{1}{2\sigma^2}\|\theta\|^2)$
This convex function can be optimized by a number of techniques such as the Quasi-Newton optimization methods. Specially for discrete-valued chain models, the observation dependent normalization can be efficiently computed by tensor/matrix multiplication Sminchisescu et al. (2005).

4.3 Action Recognition

So far what we have is a labeled data sequence $\mathcal{D} = \{\mathbf{R}^i, \mathbf{X}^i\}_{i=1}^{N}$, and we have computed the CRF model parameters θ^*. Now, once we have a new test sequence $\mathbf{x}$, we perform the same task as defined above in Section 3: we extract the foreground, divide it into two parts and compute the self-similarities. We then convert these self-similarities to HoG descriptors. Once we have these descriptors, we are ready to determine the test sequence's correct class assignment. What we want to do is to to estimate the most probable sequence label $\mathbf{r}^*$ that maximizes the conditional model.

$$\mathbf{r}^* = \arg\max_{\mathbf{r}} P(\mathbf{r}|\mathbf{x},\theta^*) \qquad (7)$$

where the parameters θ^* are learned from the training samples. While another option could be to use Viterbi path, in our experiments we the above maximal marginal probabilities for training, and the Viterbi path for labeling a new sequence for performing action recognition.

	bend	wave1	wave2	pjump	skip	jack	jump	run	walk	side
bend	**66.7**	11.1	0.0	0.0	0.0	22.2	0.0	0.0	0.0	0.0
wave1	22.2	**55.6**	22.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0
wave2	0.0	0.0	**66.7**	22.2	0.0	0.0	0.0	0.0	0.0	11.1
pjump	0.0	0.0	0.0	**66.7**	0.0	0.0	0.0	0.0	0.0	33.3
skip	0.0	0.0	0.0	10.0	**20.0**	0.0	20.0	30.0	20.0	0.0
jack	11.1	0.0	0.0	0.0	0.0	**88.9**	0.0	0.0	0.0	0.0
jump	0.0	0.0	0.0	11.1	11.1	0.0	**66.7**	0.0	11.1	0.0
run	0.0	0.0	0.0	0.0	10.0	10.0	0.0	**60.0**	20.0	0.0
walk	0.0	0.0	0.0	0.0	0.0	0.0	0.0	30.0	**70.0**	0.0
side	11.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**88.9**

Fig. 6. Cross-validation results for action recognition of the Weizman dataset when the whole foreground patch is used for computing the self-similarity surface. The dataset contains ninety-three action sequences videos of varying lengths, each having a resolution of 180 × 144. The first row shows label for each action type, the second row show a sample frame from an action sequence, while the last row shows the extracted silhouette from the action sequence.

For all recognition experiments in the next section, we report results for n-fold cross-validation and make sure the actions of the same person do not appear in the training and in the test sets simultaneously.

5. Experimental results

In this section, we put the proposed method of using the self-similarities for action recognition to test. For this reason, we validate our approach on the publicly available Weizman dataset and compare our results with some of the other significant work done in the area. Some instances from the data set are shown in Fig. 5.

5.1 Experiments with Weizman actions dataset

To asses the discriminative power of our method on real video sequences we apply it to the standard single-view video dataset with nine classes of human actions performed by nine subjects Gorelick et al. (2007)(see Fig. 5(top)). The dataset contains ninety-three action sequences videos of varying lengths, each having a resolution of 180 × 144. The nine difference action are: `bending down`, `jumping back`, `jumping`, `jumping in place`, `galloping sideways`, `running`, `walking`, `waving one hand` and `waving two hands`. Using

	bend	wave1	wave2	pjump	skip	jack	jump	run	walk	side
bend	**77.8**	0.0	22.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0
wave1	11.1	**55.6**	33.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0
wave2	11.1	11.1	**55.6**	0.0	0.0	22.2	0.0	0.0	0.0	0.0
pjump	11.1	0.0	0.0	**77.8**	0.0	0.0	0.0	0.0	0.0	11.1
skip	0.0	0.0	0.0	10.0	**30.0**	0.0	0.0	30.0	20.0	10.0
jack	11.1	0.0	0.0	0.0	0.0	**88.9**	0.0	0.0	0.0	0.0
jump	0.0	0.0	0.0	0.0	11.1	0.0	**66.7**	11.1	0.0	11.1
run	0.0	0.0	0.0	0.0	30.0	10.0	0.0	**50.0**	10.0	0.0
walk	0.0	0.0	0.0	0.0	10.0	0.0	0.0	0.0	**90.0**	0.0
side	0.0	0.0	0.0	11.1	0.0	0.0	0.0	0.0	22.2	**66.7**

Fig. 7. Cross-validation results for action recognition of the Weizman dataset when the self-similarity surfaces are created based on the optical flows computed between consecutive frames of the action sequences.

the extracted silhouettes, we compute the self-similarity surfaces of the foreground and then transform them into HoG features for action learning.

We have described above that the foreground is divided into two different parts and then we compute the self-similarity surfaces. However, we performed experiment with dividing the foreground into different parts rather than just two. For example, the results for the case when the whole foreground object is used for computing the self-similarity surface is shown in Fig. 6. As can be seen in the figure, the action recognition results for n-fold cross-validation is almost 66%. In addition, we also test on the self-similarity surfaces that are computed based on the optical flow computed between consecutive frame of the action sequence. The confusion matrix for this testing is shown in Fig. 7. The accuracy for this approach reaches 66%. Tests were also performed on dividing the foreground into three and six parts, but no improvement in the accuracy was observed.

However, our experiments show that the best results are obtained when the foreground is divided into the top and the bottom part and using only the color information. Results for the n-fold cross-validation for this case are depicted in Fig. 8. As the confusion matrix shows, the accuracy reached for our method is 70%.

These accuracy results are very encouraging, specially since we are using very sparse descriptors for the pose (just two per frame). Although higher accuracy results have been reported Ikizler & Duygulu (2007), accuracy of the proposed method is comparable to the well known

	bend	wave1	wave2	pjump	skip	jack	jump	run	walk	side
bend	**100.0**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
wave1	12.0	**67.0**	0.0	0.0	0.0	12.0	0.0	0.0	12.0	0.0
wave2	0.0	23.0	**67.0**	0.0	0.0	12.0	0.0	0.0	0.0	0.0
pjump	12.0	0.0	0.0	**34.0**	12.0	23.0	0.0	0.0	0.0	23.0
skip	10.0	0.0	0.0	0.0	**40.0**	0.0	10.0	40.0	0.0	0.0
jack	0.0	0.0	0.0	0.0	0.0	**100.0**	0.0	0.0	0.0	0.0
jump	0.0	0.0	0.0	0.0	12.0	0.0	**67.0**	0.0	23.0	0.0
run	0.0	0.0	0.0	0.0	30.0	0.0	0.0	**60.0**	10.0	0.0
walk	0.0	0.0	0.0	0.0	0.0	0.0	10.0	0.0	**90.0**	0.0
side	0.0	0.0	0.0	0.0	0.0	34.0	0.0	0.0	0.0	**67.0**

Fig. 8. Cross-validation results for action recognition of the Weizman dataset when the foreground patch is divided into an upper and a lower part for computing the self-similarity surface.

work of Niebles et al. (2006) and also with the recently reported results in Resendiz & Ahuja (2008) posted for the same dataset.

6. Conclusion

We propose a *novel* usage self-similarity surfaces for action recognition. These surfaces are computed on an extracted foreground of a person performing an action. In contrast to Shechtman & Irani (2007); Niebles et al. (2006); Resendiz & Ahuja (2008), we compute only a few of these surfaces per frame, in fact just two features per frame. Experimental validation on Weizman datasets confirms the stability and utility of our approach. The proposed method does not rely on the structure recovery nor on the correspondence estimation, but makes only mild assumptions about the rough localization of a person in the frame. This lack of strong assumptions is likely to make our method applicable to action recognition beyond controlled datasets when combined with the modern techniques for person detection and tracking.

7. References

Ahmad, M. & Lee, S. (2006). HMM-based human action recognition using multiview image sequences, *Proc. ICPR*, pp. I:263–266.

Ali, S., Basharat, A. & Shah, M. (2007). Chaotic invariants for human action recognition, *Proc. ICCV*.

Benabdelkader, C., Cutler, R. & Davis, L. (2004). Gait recognition using image self-similarity, *EURASIP J. Appl. Signal Process.* **2004**(1): 572–585.

Bobick, A. & Davis, J. (2001). The recognition of human movement using temporal templates, *PAMI* **23**(3): 257–267.

Brand, M., Nuria, O. & Pentland, A. (1997). Coupled hidden markov models for complex action recognition, *Proc. CVPR*.

Bregonzio, M., Gong, S. & Xiang, T. (2009). Recognising action as clouds of space-time interest points, *Proc. CVPR*, pp. 1948–1955.

Carlsson, S. (2000). Recognizing walking people, *Proc. ECCV*, pp. I:472–486.

Carlsson, S. (2003). Recognizing walking people, *I. J. Robotic Res.* **22**(6): 359–370.

Cutler, R. & Davis, L. (2000). Robust real-time periodic motion detection, analysis, and applications, *PAMI* **22**(8): 781–796.

Dalal, N. & Triggs, B. (2005). Histograms of oriented gradients for human detection, *Proc. CVPR*, pp. I:886–893.

Fathi, A. & Mori, G. (2008). Action recognition by learning mid-level motion features, *Proc. CVPR*.

Filipovych, R. & Ribeiro, E. (2008). Learning human motion models from unsegmented videos, *Proc. CVPR*.

Gilbert, A., Illingworth, J. & Bowden, R. (2008). Scale invariant action recognition using compound features mined from dense spatio-temporal corners, *Proc. ECCV*, pp. I: 222–233.

Gorelick, L., Blank, M., Shechtman, E., Irani, M. & Basri, R. (2007). Actions as space-time shapes, *PAMI* **29**(12): 2247–2253.

Ikizler, N., Cinbis, R. G. & Duygulu, P. (2008). Human action recognition with line and flow histograms, *Proc. ICPR*.

Ikizler, N. & Duygulu, P. (2007). Human action recognition using distribution of oriented rectangular patches, *Workshop on Human Motion*, pp. 271–284.

Ikizler, N. & Forsyth, D. (2007). Searching video for complex activities with finite state models, *Proc. CVPR*.

Jhuang, H., Serre, T., Wolf, L. & Poggio, T. (2007). A biologically inspired system for action recognition, *Proc. ICCV*.

Jia, K. & Yeung, D.-Y. (2008). Human action recognition using local spatio-temporal discriminant embedding, *Proc. CVPR*.

Jingen, L., Saad, A. & Shah, M. (2008). Recognizing human actions using multiple features, *Proc. CVPR*.

Lafferty, J., McCallum, A. & Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data, *In Proceedings of the Eighteenth International Conference on Machine Learning (ICML)*.

Laptev, I. & Lindeberg, T. (2003). Space-time interest points, *Proc. ICCV*, pp. 432–439.

Li, R., Tian, T. & Sclaroff, S. (2007). Simultaneous learning of nonlinear manifold and dynamical models for high-dimensional time series, *Proc. ICCV*.

Lin, Z., Jiang, Z. & Davis, L. S. (2009). Recognizing actions by shape-motion prototype trees, *Proc. ICCV*.

Liu, J. & Shah, M. (2008). Learning human actions via information maximization, *Proc. CVPR*.

Lucas, B. & Kanade, T. (1981). An iterative image registration technique with an application to stereo vision, *Image Understanding Workshop*, pp. 121–130.

Lv, F. & Nevatia, R. (2007). Single view human action recognition using key pose matching and viterbi path searching, *Proc. CVPR*.

Mikolajczyk, K. & Uemura, H. (2008). Action recognition with motion-appearance vocabulary forest, *Proc. CVPR*, pp. 1–8.

Moeslund, T., Hilton, A. & Krüger, V. (2006). A survey of advances in vision-based human motion capture and analysis, *CVIU* **103**(2-3): 90–126.

Natarajan, P. & Nevatia, R. (2008). View and scale invariant action recognition using multiview shape-flow models, *Proc. CVPR*, pp. 1–8.

Niebles, J., Wang, H. & Li, F. (2006). Unsupervised learning of human action categories using spatial-temporal words, *Proc. BMVC*.

Ogale, A., Karapurkar, A. & Aloimonos, Y. (2006). View-invariant modeling and recognition of human actions using grammars, *Proc. Workshop on Dynamic Vision*, pp. 115–126.

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

Rao, C., Yilmaz, A. & Shah, M. (2002). View-invariant representation and recognition of actions, *IJCV* **50**(2): 203–226.

Rapantzikos, K., Avrithis, Y. & Kollias, S. (2009). Dense saliency-based spatiotemporal feature points for action recognition, In Proc. CVPR.

Resendiz, E. & Ahuja, N. (2008). A unified model for activity recognition from video sequences, *Proc. ICPR*, pp. 1–4.

Shechtman, E. & Irani, M. (2007). Matching local self-similarities across images and videos, *Proc. CVPR*.

Shen, Y. & Foroosh, H. (2008). View invariant action recognition using fundamental ratios, *Proc. CVPR*.

Sminchisescu, C., Kanaujia, A., Li, Z. & Metaxas, D. (2005). Conditional models for contextual human motion recognition, *Proc. CVPR*.

Syeda-Mahmood, T., Vasilescu, M. & Sethi, S. (2001). Recognizing action events from multiple viewpoints, *Proc. EventVideo*, pp. 64–72.

Turaga, P. K., Chellappa, R., Subrahmanian, V. S. & Udrea, O. (2008). Machine recognition of human activities: A survey, *IEEE Trans. Circuits Syst. Video Techn.* **18**(11): 1473–1488.

Wang, L., Hu, W. & Tan, T. (2003). Recent developments in human motion analysis, *Pattern Recognition* **36**(3): 585–601.

Weinland, D. & Boyer, E. (2008). Action recognition using exemplar-based embedding, *Proc. CVPR*.

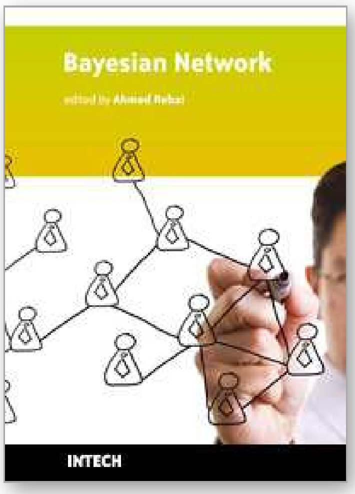
Weinland, D., Ronfard, R. & Boyer, E. (2006). Free viewpoint action recognition using motion history volumes, *CVIU* **103**(2-3): 249–257.

Wilson, A. & Bobick, A. (1995). Learning visual behavior for gesture analysis, *IEEE Symp. on Comp. Vision*.

Yeffet, L. & Wolf, L. (2009). Local trinary patterns for human action recognition, *Proc. ICCV*.

Yilmaz, A. & Shah, M. (2005a). Actions sketch: A novel action representation, *Proc. CVPR*, pp. I:984–989.

Yilmaz, A. & Shah, M. (2005b). Recognizing human actions in videos acquired by uncalibrated moving cameras, *Proc. ICCV*, pp. I:150–157.



Bayesian Network
Edited by Ahmed Rebai

ISBN 978-953-307-124-4
Hard cover, 432 pages
Publisher Sciyo
Published online 18, August, 2010
Published in print edition August, 2010

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821