Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/090636

Human Activity Recognition: Discriminative Models using Statistical Chord-length and Optical Flow Motion Features

Mahmoud Elmezain[1,2,*] *and Essam O. Abdel-Rahman*[1,3]

[1] Faculty of Science and Computer Engineering, Taibah University, Yanbu, KSA.
[2] Faculty of Science, Tanta University, Tanta, Egypt.
[3] Faculty of Computers and Information, Minia University, Minia, Egypt.

Received: 4 Mar. 2015, Revised: 2 May 2015, Accepted: 3 May 2015
Published online: 1 Nov. 2015

Abstract: Despite the vast amount of research on the analysis of existing and ongoing human activity, there are still significant challenges worthy of address. In this paper, an innovative approach for human action recognition based on discriminative models like CRFs, HCRFs and LDCRFs is proposed. To handle human action recognition, different number of window size ranging from 0 to 7 are applied using a compact computationally-efficient descriptor as statistical chord-length features (SCLF), in addition to optical flow motion features that derived from 3D spatio-temporal action volume. Our experiment on a standard benchmark action KTH, as well as our IIKT dataset show that the recognition rate, and the reliability of human activity is improved initially as the window size increase, but degrades as the window size increase further. Furthermore, LDCRFs is robust and efficient than CRFs and HCRFs, in addition to problematic phenomena than those previously reported. It also can carry out without sacrificing real-time performance for a wide range of practical action applications.

Keywords: Action Recognition, Statistical Chord-length Features, Discriminative Models.

1 Introduction

Automatically recognizing human activities into video sequences is increasingly receiving research attentions due to its great potentials for many applications in several contexts and domains [1]. For example, action-based Human Computer Interaction (HCI) is probably one of the most widespread applications for human action recognition, where no explicit actions as keystrokes and mouse clicks are available to capture user input. Many approaches deem an action as a sequence of observations. For this view, an activity is represented by a sequence of feature vectors picked up from video data; thus by searching for such sequence, the activity can be recognized. One notice via literature scanning that a significant working body in action recognition focuses on using feature descriptors and spatial-temporal key points. Generally, there are several existing surveys within the area of human action that can be classified using various visual cues such as shape [2] and motion [3].

In [4], the authors present an approach to represent and recognize the human movement. In their work, a representation known as ”temporal templates” are introduced to capture both motion and shape, represented as evolving silhouettes. Two 2D images; motion energy images and motion history images, instead of maintaining 3D spatio-temporal volumes, are employed as templates for action recognition. In [5], the authors present an approach that extracts spatio-temporal features at multiple temporal scales to isolate and cluster actions. To deal with the speed variations of actions, they analyze manifold temporally scaled video volumes. Then local intensity gradients are estimated and normalized for all points within a 3D volume. *Shectman* and *Irani* proposed an approach to estimate motion flows for recognizing human action from a 3D spatio-temporal correlation volume that detects similarities among video segments [6]. *Ahmad* and *Lee* presented a method for human action recognition from multivites image sequences, which uses the integrated shape and motion flow information with

[*] Corresponding author e-mail: Mahmoud.Elmezain@tuscs.com

variability consideration [7]. In this method, a set of multi-dimensional combined local-global optic flow and shape flow feature vectors are employed for a set of multi-dimensional Hidden Markov Model (HMM) for modeling human action.

The major problem that arises here is that the comparison of results obtained with different datasets can be difficult. For this reason and to avoid this problem, many other researchers [8,9,10,11,12,13,14,15,16,17], have preferred to use some common datasets to evaluate their systems effectiveness. In this case, the comparison with other recognition methods turns out to be very meaningful and just fair, as all techniques use the same public dataset and the same experimental settings. In the literature, there are a variety of benchmark datasets (e.g., KTH [18] and Weizmann [19], etc.) commonly used to evaluate activity recognition algorithms. These datasets differ notably from one to another in many aspects (e.g., the number of action categories, the number of actions per category, the number of subjects performing actions, camera viewpoints, illumination, occlusion, etc.).

Ke *et al.* proposed a novel appearance-based framework that employs volumetric features for efficiently analyzing video's optical flow [8]. This framework extends the rectangle feature into spatio-temporal domain, in addition to sperate the optical flow into the horizontal and vertical components and compute volumetric features on each component. *Fathi* and *Mori* developed an approach for action recognition based on mid-level motion features, which are built from low-level optical flow information and classified by a binary AdaBoost classifier [14]. In [2], the authors proposed an innovative approach for human activity recognition based on affine-invariant shape representation and Support Vector Machine (SVM) based feature classification. Sminchisescu *et al.* [20] applied CRFs model to recognize human motion activities and showed improvement over the Hidden Markov Models (HMMs) technique.

The main contribution in this paper is to investigate humane action recognition based on an affine-invariant shape descriptor like SCLF, in addition to mass center and optical flow motion features. The extracted features from 3D spatio-temporal volumes are employed with varying windows size for the discriminative models as CRFs, HCRFs and LDCRFs to recognize the human activities in image sequences. Our experiments on standard benchmark action KTH and IIKT datasets show that the proposed approach is more robust and yields promising results when comparing favorably with those previously reported throughout the literature without sacrificing real-time performance. The rest of this paper is organized as follows; Section 2 demonstrates the literature review. Section 3 roads the systematic concept of the human action recognition approach in three subsections. Experimental results on human actions are described in Section 4. Finally, Section 5 summaries and concludes this paper.

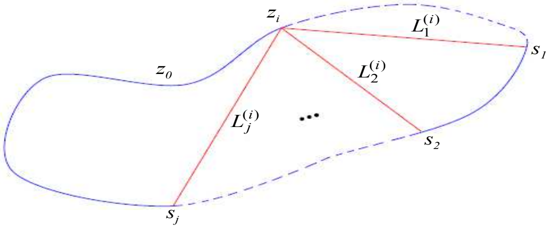

Fig. 1: An example of CLFs obtained through the division of a shape contour into a finite number of arcs of equal length.

2 Related Literature

A good choice for classification approaches helps the success of any system and makes it suitable for real-world applications. In this paper, human actions are classified according to discriminative models like CRFs, HCRFs and LDCRFs, which enforce the vigorous view-invariant task. So, this section is important in the context of understanding the motivation of doing the research and enables to investigate and compare the novel techniques. The following two subsections briefly review the statistical chord-length function and the Conditional Random Fields classifier.

2.1 Chord-Length Function

The chord-length shape features are constructed using 1-D chord-length functions. Formally speaking, the contour $\mathscr{C}$ of a 2-D shape can be defined as an ordered sequence of N coordinate points [21,22];

$$\mathscr{C} = \{z_t = (x_t, y_t) \in R^2 | t = 0, 1, ..., N-1\} \tag{1}$$

where $z_{t+N} = z_t$ as $\mathscr{C}$ is closed. The diameter D of the shape is given by;

$$D = \max_{i,j=0}^{N-1} \|z_i - z_j\|, \quad i \neq j \tag{2}$$

where $\|.\|$ is defined as the Euclidean distance between two points z_i and z_j. Let us take a point $z_i \in \mathscr{C}$, as starting point and the contour $\mathscr{C}$ be traversed anti-clockwise and divided into $k > 1$ sections, i.e.,$\widehat{z_i s_1}, \widehat{s_1 s_2}, ..., \widehat{s_{k-1} z_i}$ of equal length, where s_j is the j^{th} division point and $1 \leq j < k$. Thereby, we have $k-1$ chords having the lengths;

$$L_1^{(i)}, L_2^{(i)}, ..., L_{k-1}^{(i)} \tag{3}$$

where $L_j^{(i)}$ is the length of the chord $\widehat{z_i s_j}$ that measured as the Euclidean distance between the two points s_j and z_i, as illustrated in Fig. 1.

Let us now show that while the point z_i moves along the contour, the chord length's $L_j^{(i)}$ will vary accordingly. This implies that $L_j^{(i)}$ is a function of z_i. Here a function is called the Chord-Length Function (CLF), and shortly denoted as $L_j^{(i)}$. Thus we can get $k-1$ CLFs, i.e., $L_1, L_2, ..., L_{k-1}$. Since these functions are obtained from splitting the contour evenly and from moving the initial point z_i, along the contour, so that they guarantee to be invariant to translation and rotation. However, the chord length itself is not a scale invariant, but it can be made to be invariant to scale by normalization using the contour diameter D. The CLFs apparently meet the key requirements for being a shape descriptor. Then we need to scale all the CLFs to be within the same range (e.g. [0, 1]). By their definition, CLFs are obtained by segmenting the contour evenly, so that it is easy to deduce that only half of the CLFs, $L_1, L_2, ..., L_{k/2}$ are enough to describe the shape adequately. It is germane to point to the fact that both global and local features of shape can be captured by using chord-lengths of different level. This is viewed as a distinct competitive advantage of the CLF-based descriptor over other shape descriptors.

2.2 Conditional Random Fields

Conditional Random Fields are undirected graphical models that were developed for labeling sequential data [23]. However, each label (state) corresponds to a specific human action. Moreover, there is a trade-off in the weights of each feature function for each state because CRFs use a single exponential distribution to model all reference labels of given observation [24]. The CRFs are satisfied by defining the normalized each product of potential function. In the case of chain-structured CRFs as depicted in Fig. 2, each potential function operates on pairs of adjacent label variables y_i and y_{i+1}.

Formally speaking, for each observation sequence $x = \{x_1, x_2, ..., x_m\}$ such that each frame observation x_j is represented by a feature vector $\phi(x_j) \in R^d$ and a label y that is a member of a set $\mathscr{Y}$ of possible class labels, the probability of label sequence y given observation sequence x is calculated as;

$$p(y|x,\theta) = \frac{1}{Z(x,\theta)} \exp\left(\sum_{i=1}^{n} F_\theta(y_{i-1}, y_i, x, i)\right) \tag{4}$$

where parameter $\theta = (\lambda_1, \lambda_2, ..., \lambda_{N_f}; \mu_1, \mu_2, ..., \mu_{N_g})$, N_f represents the number of transition feature function, N_g refers to the number of state feature function and n is the length of observation sequence x. F_θ is defined as;

$$F_\theta(y_{i-1}, y_i, x, i) = \sum_f \lambda_f t_f(y_{i-1}, y_i, x, i) + \sum_g \mu_g s_g(y_i, x, i) \tag{5}$$

where $t_f(y_{i-1}, y_i, x, i)$ is a transition feature function at position i and $i-1$. $s_g(y_i, x, i)$ refers to a state feature function at position i. λ_f and μ_g represent the weights of the transition and the state feature functions, respectively. $Z(x,\theta)$ is the normalized factor where it is calculated as follows;

$$Z(x,\theta) = \sum_y \exp\left(\sum_{i=1}^{n} F_\theta(y_{i-1}, y_i, x, i)\right) \tag{6}$$

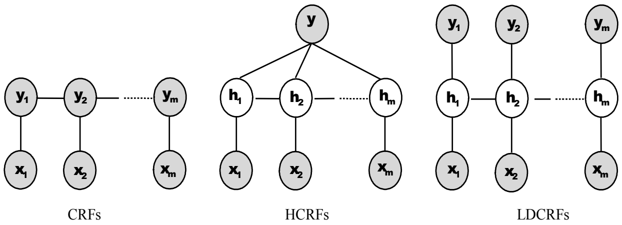

Fig. 2: Different type of discriminative models: CRFs, HCRFs and LDCRFs. In these models, x_j refers to the j^{th} corresponding observation value, h_j is a hidden states that assigned to x_j. y_j is the label of x_j where the gray circles represent the observed variables.

Other approaches, including the hidden variables offer several advantages over previous CRFs model. Although the CRFs model the transition among actions and overcome the weakness of directed graphical models, which suffer from bias problem, it does not have the ability to learn the internal sub-structure of action sequences. Hidden Conditional Random Fields (HCRFs) are the extension of CRFs that include hidden variables [25,26]. HCRFs can automatically model the local interconnection between labels (i.e. states) with hidden variables, but it cannot model dynamics among states. On the other sides, Latent-Dynamic Conditional Random Fields (LDCRFs) can model the sub-structure of a state and learn dynamic among states [27]. The LDCRFs model combines the strengths of CRFs and HCRFs. Furthermore, it can detect and recognize states from un-segment data (Fig.2).

3 Proposed Methodology

In this section, the proposed approach for action recognition is described. The main steps within the framework are explained in detail along the following subsections (Fig. 3).

3.1 Preprocessing

Background subtraction is a widely used approach for detecting the unusual motion in a scene, which involves comparing each new frame to a designed model against the scene background. It is worth mentioning that, Gaussian Mixture Models (GMM) are an example of a larger class of density models that have several functions as additive components [28].

Formally speaking, Let X_t be a pixel in the current frame, and K is the number of distributions. Thus, each

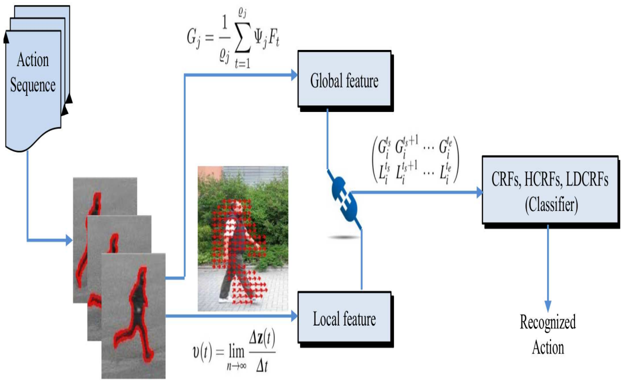
Action Sequence
$G_j = \frac{1}{\varrho_j}\sum_{t=1}^{\varrho_j}\Psi_j F_t$
Global feature
$\begin{pmatrix} G_i^{t_s} & G_i^{t_s+1} & \cdots & G_i^{t_e} \\ L_i^{t_s} & L_i^{t_s+1} & \cdots & L_i^{t_e} \end{pmatrix}$
CRFs, HCRFs, LDCRFs (Classifier)
Local feature
$\upsilon(t) = \lim_{n\to\infty}\frac{\Delta \mathbf{z}(t)}{\Delta t}$
Recognized Action

Fig. 3: Road map of the proposed approach for action recognition.

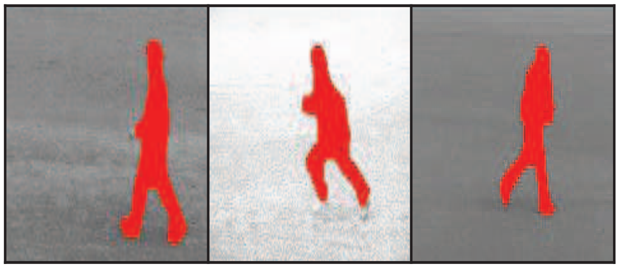

Fig. 4: Foreground detection via background subtraction.

pixel can be modeled separately by a mixture of K Gaussian as follows;

$$p(X_t) = \sum_{i=1}^{K} \omega_{i,t} \cdot \eta(X_t; \mu_{i,t}; \Sigma_{i,t}) \tag{7}$$

where η refers to a Gaussian probability density function. $\mu_{i,t}, \Sigma_{i,t}$ and $\omega_{i,t}$ represent the mean, covariance and an estimate of the prior probability of the i^{th} component (i.e. weighting function), respectively.

In our work, a constructive algorithm, which uses the criteria of maximizing a likelihood function is employed to decide the number of components automatically [28]. Additionally, the background is updated and optimized based upon the minimization of error function E (Eq. 8).

$$E = -\sum_{n=1}^{N} ln\Big(\sum_{i=1}^{K} \eta(X; \mu_i; \Sigma_i) \cdot \omega_i\Big) \tag{8}$$

where N represents the number of data points X_n. By applying a threshold γ = 0.5, the background distribution remains on top with the lowest variance. Finally, all pixels X that match none of the components are best candidates to be marked as foreground. An example from the results of background estimation with K = 5 is shown in Fig. 4. For more details, the reader can refer to [29].

3.2 Feature Extraction

In this approach, a variety of local and global features are used to describe the segmented silhouettes of moving human body parts $f(x,y,t)$. For shape (global) features, the silhouette image sequences are considered with an invariant descriptor such as statistical chord-length feature. Additionally, the motion (local) features of foreground image sequence which are extracted by the trajectory of the motion centroid and optic flows are also used. Thereby, the feature matrix of human action is represented by the following equation;

$$Action_{features} = \begin{pmatrix} G_i^{t_s} & G_i^{t_s+1} & \cdots & G_i^{t_e} \\ L_i^{t_s} & L_i^{t_s+1} & \cdots & L_i^{t_e} \end{pmatrix} \tag{9}$$

where G_i and L_i refer to global and local feature, respectively. The duration of feature's action is the difference between starting frame (t_s) and ending frame (t_e), where $G_i^{t_s}$=$\{g_1^{t_s}, g_2^{t_s}, ..., g_{F_1}^{t_s}\}$ and $L_i^{t_s}$=$\{l_1^{t_s}, l_2^{t_s}, ..., l_{F_2}^{t_s}\}$. Thus, the combined feature of global and local features at each frame is F_1+F_2.

3.2.1 Global Feature

Although the chord-length function has an ability to be invariant with respect to translation, rotation, and scaling, but it appears to be not compact enough. In addition, the chord-length function may be changed because it constantly depends on a reference point whereby the shape border is parameterized. The reason of its dependence is that the contour is closed, and any point on the contour can be used as a reference point. To alleviate such problems and for convenience, the mean μ_j and variance σ_j of the chord-length functions L_j, j = 1, 2, ..., k/2 are estimated as;

$$\mu_j = \frac{1}{N}\sum_{i=0}^{N-1} L_j^{(i)}, \quad \sigma_j = \frac{1}{N-1}\sum_{i=0}^{N-1}(L_j^{(i)} - \mu_j)^2 \tag{10}$$

Therefore, the chord-length features that are used as a shape descriptor can be arranged in 1-D of size k as follows;

$$F = (\mu_1, \sigma_1, \mu_2, \sigma_2, ..., \mu_{k/2}, \sigma_{k/2})^T \tag{11}$$

To get the final chord-length features of a given human action, we firstly obtain the chord-length features of all pose of that action. Shortly speaking, each action snippet is temporally divided into a number of fuzzy states, each represents a pose of the action, then the chord-length features of an action pose is occurred by;

$$G_j = \frac{1}{\rho_j}\sum_{t=1}^{\rho_j} \Psi_j F_t, \quad j = 1, 2, ..., m \tag{12}$$

where $\Psi_j \in [0,1]$ is the fuzzy membership function that defines the temporal slice j, ρ_j is the total number of the chord-length feature vectors of the pose j, and m is the total number of time-slices. Accordingly, the final feature vector of a given action can be constructed by catenating all the descriptors of its temporal poses. The resulting feature vectors are normalized to the integral value of unity to achieve robustness to scale variations and to reduce the influence of illumination. The normalized feature vectors obtained can now be exploited as shape descriptors for action classification and recognition.

3.2.2 Local Feature

The motion flow as the local flow of foregrounds is characterized and stated by the center of gravity and optic flow L_i=$[\mathbf{z}(t), \upsilon_{op}]^T$ as follows described;

Center of Silhouettes Motion (CM) The use of motion information motivates us to fuse it with global features to form the final CRFs classifier. The motion features extracted here are based on calculating the centroid $\mathbf{z}(t)$ that delivers the center of motion. Therefore, the features $\upsilon(t)$ describing the general distribution of motion are given by;

$$\upsilon(t) = \lim_{n\to\infty} \frac{\Delta \mathbf{z}(t)}{\Delta t} \tag{13}$$

where $\frac{1}{2}\left(\sum_{i=1}^{n} x_i, \sum_{i=1}^{n} y_i\right)$ are the spatial coordinates of $\mathbf{z}(t)$ with respected to the total number of moving pixels n in the given frame. Such features have profound implications, not only about the type of motion (e.g., translationally or oscillatory), but also about the rate of motion (i.e. velocity). With these features, it would be able to distinguish, for example, between an action where motion occurs over a relatively large area (e.g., running) and an action localized in a smaller region, where only small parts of the body are in motion (e.g., waving either one or two hands). It is worth mentioning that fusing motion information with regular global features consistently boosts action recognition (i.e., leads to an overall increase in recognition rates).

Optic Flow It is being notice that related body parts involve optical flow velocity. As, the person conducts the action ”hand waving”, motion only involves the hand. However, when the person conducts the action ”walking”, the motion involves the whole body. Furthermore, pruning of computed flow values appears to be a clue to accurate flow fields, which in turn allows for better motion estimation. Optical flow pruning involves two passes, each based on the magnitude (Euclidean length) of optical flow vectors to separate relevant from irrelevant flow vectors [14]. In the first pass, all flow vectors whose magnitudes are either relatively truly small or very large are removed [30].

For this purpose, two predefined minimum and maximum thresholds are used to control the filtering of flow vectors. Briefly speaking, given two thresholds ρ_1 and ρ_2, a flow vector $\upsilon_{op} = [x,y]^T$ is only accepted as valid if it satisfies the validity constraint: $\rho_1 < \|\upsilon_{op}\| < \rho_2$ where $\|.\|$ denotes the magnitude of the flow vector with respect to the Euclidean metric; otherwise, it is assumed to be a noisy flow component and thus removed. For the second pass, a vector υ_{op} is treated as a valid flow component if the Euclidean distance between the center of flow and the vector being analyzed does not exceed a specific threshold τ. Formally, this is expressed as;

$$\|\upsilon_{op} - \mathbf{z}\| < \tau \tag{14}$$

where $\mathbf{z}$ is the centroid of the motion region. In our experiments, the setting values of $\rho_1 = 5, \rho_2 = 20$ and τ at 25% of the average of image width w and height h; $\ell = (w+h)/2$ give an overall good pruning performance.

3.3 Classification

Throughout the classification stage, the action recognition is handled according to three classifiers: CRFs, HCRFs and LDCRFs to decide which one is the best in terms of performance. The action recognition module matches the tested human activity against the reference database, to classify which class it belongs to. Thereby, the human action sequence is recognized corresponding to the maximal likelihood of all actions (i.e. labels) accumulatively. The maximal label of CRFs model is the action whose observation probability is the largest among all the action’s labels. The following two subsections briefly review the learning and the inferencing CRFs.

3.3.1 Learning CRFs Model

The parameter $\theta = (\lambda_1, \lambda_2, ..., \lambda_{N_f}; \mu_1, \mu_2, ..., \mu_{N_g})$ is determined from training data $D = \{(x^{(j)}, y^{(j)})\}_{j=1}^{T_d}$, where $x^{(j)}$ is an observation sequence of training set, $y^{(j)}$ is the corresponding label sequence for observation sequence $x^{(i)}$ and T_d is the number of training sequences. The objective function to learn the parameter θ that maximize the log-likelihood of training data is computed by;

$$\begin{aligned} L(\theta) &= \sum_{j=1}^{T_d} \log p(y^{(j)}|x^{(j)}, \theta) \\ &= \sum_{j=1}^{T_d} \left(\sum_{i=1}^{n} F_\theta(y_{i-1}^{(j)}, y_i^{(j)}, x^{(j)}, i) - \log Z(x^{(j)}, \theta) \right) \end{aligned} \tag{15}$$

Likelihood maximization can be performed using a gradient ascent method the BFGS optimization technique of 300 iterations to converge [31] :

$$\frac{\partial L(\theta)}{\partial \theta} = \sum_{j=1}^{T_d} \Big(\sum_{i=1}^{n} \frac{\partial F_\theta(y_{i-1}^{(j)}, y_i^{(j)}, x^{(j)}, i)}{\partial \theta} - \sum_{x} p(y|x^{(j)}) \sum_{i=1}^{n} \frac{\partial F_\theta(y_{i-1}, y_i, x^{(j)}, i)}{\partial \theta} \Big) \qquad (16)$$

Based on the above-mentioned steps, HCRFs and LDCRFs models have a similar computational complexity to fully observable CRFs.

3.3.2 Inference CRFs Model

To compute the probability $p(y|x,\theta)$ of label sequence y given a new observation sequence x, a set of matrices is computed [20,23,32]. To simplify some expressions, special start y_0 and stop y_{n+1} states are added. These states are dummy. Suppose that $p(y|x,\theta)$ is given by 5. For each position i in the observation sequence, $M_i(x)$ is $|\mathscr{Y} \times \mathscr{Y}|$ matrix is defined as follows;

$$M_i(y', y|x) = \exp\left(F_\theta(y', y, x, i)\right) \qquad (17)$$

where $\mathscr{Y} = \{y_1, y_2, ..., y_l\}$ is a set of label of the training data. l is the number of the labels. y' and y are the labels of S at time i. Using this notation, the conditional probability of a label sequence y is computed as;

$$p(y|x,\theta) = \frac{\prod_i^{n+1} M_i(y_{i-1}, y_i|x)}{Z(x,\theta)} \qquad (18)$$

The normalization $Z(x,\theta)$ is the entry of product of these matrices:

$$Z(x,\theta) = \Big(\prod_{i=1}^{n+1} M_i(i) \Big)_{start,stop} \qquad (19)$$

4 Experimental Results

To evaluate the proposed approach, two main experiments were carried out, and the results we achieved were compared with those reported by other state-of-the-art methods. In order to provide an unbiased estimation of the generalization abilities of the classification process, the sequences, for each action, were divided into a training set (two thirds) and a test set (one third). In this work, CRFs, HCRFs and LDCRFs are trained using gradient ascent with the BFGS optimization technique with 300 iterations to converge. The training process is more expensive ranging from 20 minutes to several hours for models having longer windows of observations on a standard desktop PC. On the contrary, inference (i.e. recognition) is about as fast for all models in the order of seconds for sequences of several frames (i.e. more than 20 frames in a sequence). In addition, inference process used forward score of each sample to select the label with the highest likelihood.

In an automatic action recognition task, there are three types of errors called insertion, substitution and deletion. The insertion error is occurred when the classifier detects a nonexistent action. It is because the emission probability of the current label for a given observation sequence is equal to zero. A substitution error occurs when the action is classified falsely (i.e. classifies the human action as another action). This error is usually happened when the extracted features are falsely employed to other features. The deletion error happens when the classifier fails to detect a meaningful action. In order to calculate the recognition ratio, insertion errors are totally not considered (Eq. 20). However, insertion errors are probably caused due to substitution and deletion errors because they are often considered as a strong decision in determining the meaningful actions.

$$Recognition\ ratio = \frac{\#\ recognized\ actions}{\#\ test\ actions} \times 100 \qquad (20)$$

Deletion errors directly affect the recognition ratio whereas insertion errors do not. However, the insertion errors affect the action recognition ratio directly. To consider the effect of insertion errors, another performance measure called reliability is estimated by the following equation;

$$Reliability = \frac{\#\ correctly\ recognized\ actions}{\#\ test\ actions + \#\ Inseration\ errors} \times 100 \qquad (21)$$

Furthermore, the action recognition accuracy is measured according to different window size ranging from 0 to 7 to decide the best in terms of recognition results. A window size of zero means that the feature matrix at the current frame is only used to construct the input feature while the window size of three means that the input feature matrix at each frame consists of seven features, which contain the current frame, three preceding frames and three future frames. So, multiple experiments have been conducted with a variety of window sizes on the proposed approach to empirically conclude the optimal outcome of the system.

4.1 Experiment 1

In real-world scenarios, we decided to create our own realistic action recognition dataset (hereinafter called as IIKT[1] action dataset) which is going to be publicly available free of restrictions on use for action recognition research on the Web very soon. This action database

[1] IIKT is an acronym for the German expression: ”Institut für Informations und Kommunikationstechnik”; the Institute for Information Technology and Communications at OvG University Magdeburg, Germany and is one of the largest engineering schools in Germany.

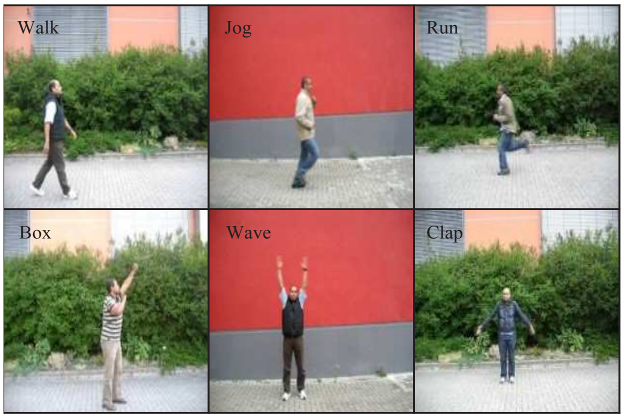

Fig. 5: Sample frames form action sequences in the IIKT dataset.

contains a total of six action categories; three ”leg actions” (i.e., walking, jogging, and running) and three ”arm actions” (i.e., boxing, hand-waving, and hand-clapping). The video sequences were typically acquired by a Canon IXUS 65 digital cameras at 30 FPS with 640 × 480 pixels image resolution represented in 256 grayscale levels. Within the sequences, actions are performed by six subjects; each subject was asked to wear a different clothing item. Each action sequence was then segmented into shorter video clips of 53 sec. duration, which we termed ”action snippets”. Fig. 5 shows example frames from action sequences of different categories represented in the IIKT dataset. In this work, a motion-related descriptor based on optical flow analysis is proposed. However, most optical flow computations turn out to be sensitive to background noise, and changes in scale and/or directionality of motion. Furthermore, the number of moving pixels is subject to change with time. Due to these restrictions, raw values of optical flow would likely be less suitable or unsuitable as features for motion analysis. In order to overcome these difficulties, the characteristics of distribution of optical flow as features to describe motion is used. As a matter of fact, one can see that the motion activity of an individual moving in a scene with a static background can be characterized fully by its own self-induced optical flow profile. In Fig. 6, samples optical flow patterns in a sequence showing a person performing actions of walking, boxing and clapping are illustrated.

According to the above-mentioned discriminative models, the human action accuracy is measured according to different sliding window sizes ranging from 0 to 7 (Fig.7). It noted that the action recognition accuracy and the reliabilities of CRFs, HCRFs and LDCRFs was improved initially as the window size increase, but degrades as a window size increase further. Generally, the optimal window size of them was assigned to 4, where multiple experiments have been conducted to empirically conclude the optimum value on the outcome of the system. From Fig.7, It is being noted that, the insertion, substitution and deletion errors decrease sharply between a window size =0 and window size=4. However, deletion, insertion and substitute errors begin to increase after window size = 4. Moreover, the system can deal with several video samples, which contain confusing situations with superior performance and low computational complexity. Experimental results with CRFs, HCRFs and LDCRFs show that the proposed approach automatically recognizes human actions at window size=4 with 89.64%, 92.14% and 96.33%, receptively. And also, the reliability of these systems at the same window size is 88.38%, 91.17% and 95.41%, respectively. As a result, LDCRFs is the best in terms of results than CRFs and HCRFs.

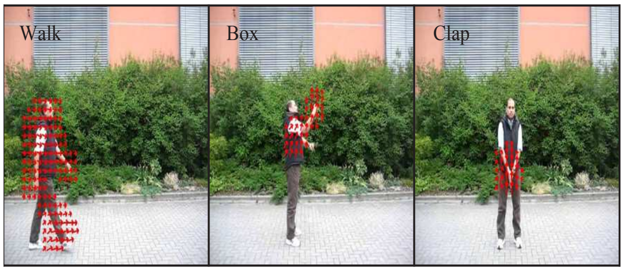

Fig. 6: Optical flow estimation results for a real-world video sequence showing a single person performing various actions, i.e. walking, boxing and clapping from left to right, respectively.

Table 1: Confusion matrix for per-video classification on IIKT dataset using LCDRFs at window size = 4.

Action	Walk	Run	Jog	Box	Wave	Clap
Walk	1.00	0.00	0.00	0.00	0.00	0.00
Run	0.00	0.94	0.06	0.00	0.00	0.00
Jog	0.00	0.05	0.95	0.00	0.00	0.00
Box	0.00	0.00	0.00	0.95	0.00	0.05
Wave	0.00	0.00	0.00	0.00	1.00	0.00
Clap	0.00	0.00	0.00	0.06	0.00	0.94

As follows from the figures tabulated in Table 1, most actions are correctly classified. Furthermore, there is a high distinction between arm actions and leg actions. Most of the mistakes where confusions occur are between ”running” and ”jogging” actions and between ”boxing” and ”clapping” actions. This is intuitively plausible due to the fact of high similarity between each pair of these actions.

4.2 Experiment 2

This second experiment was conducted using KTH dataset [18]. The KTH human action dataset includes six actions: walking, running, jogging, boxing, hand waving and hand clapping, which performed by 25 subjects (Fig. 8). Four different scenarios are used; outdoors, outdoors

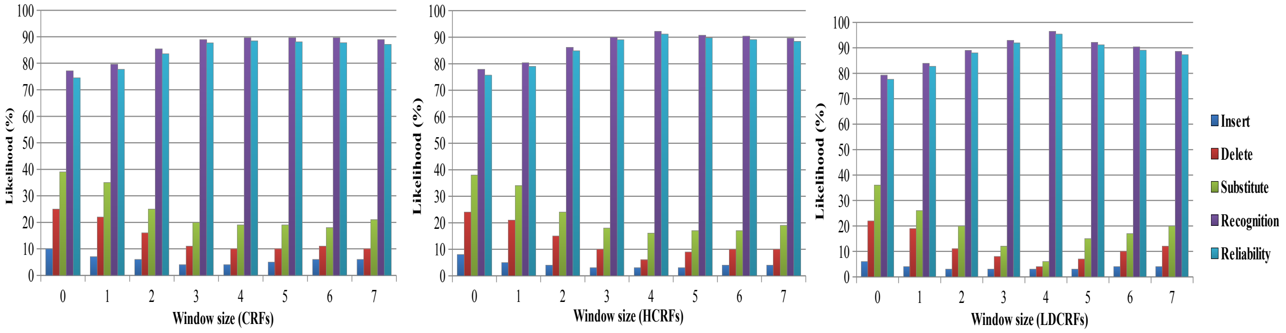

Fig. 7: Insertion, deletion and substitution errors, as well as the recognition and reliability of IIKT dataset using CRFs, HCRFs and LDCRFs relative to sliding window size ranging from 0 to 7.

with scale variation, zooming, outdoors with distinct clothes and indoors. Apart from zooming setting, there is an only slight camera a movement, variation in the performance and duration, and somewhat in the viewpoint. The backgrounds are relatively static, where all sequences are acquired at 25 FPS and a spatial resolution of 160 × 120 pixels.

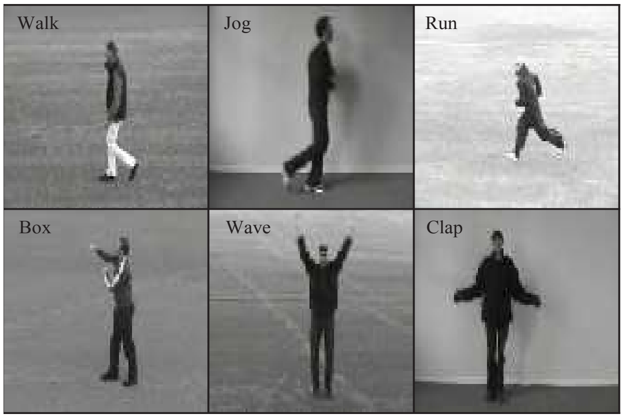

Fig. 8: Sample frames form action sequences in the KTH dataset [18].

Similarly, the human action accuracy is measured according to different sliding window sizes ranging from 0 to 7 (Fig.9). It is being observed that the action recognition accuracy and the reliabilities of CRFs, HCRFs and LDCRFs was improved initially as the window size increase, but degrades as a window size increase further. Experimental results with CRFs, HCRFs and LDCRFs show that the proposed approach automatically recognizes human actions at window size=4 with 90.36%, 93.57% and 98.50%, receptively. In addition, the reliability of these systems is 89.08%, 92.58% and 97.87%, respectively. Furthermore, it is being noted that, LDCRFs is the best in terms of results than CRFs and HCRFs, for all window size ranging from 0 to 7.

The confusion matrix depicting the results of action recognition achieved by using LDCRFs at window size = 4 is shown in Table 2. Here, there is a clear distinction between arm actions and leg actions. Most of the mistakes where confusions occur are between ”running” and ”jogging” and actions and between ”boxing” and ”clapping”. Thereby, our method has achieved a 98.50% accuracy per-video classification.

Table 2: Confusion matrix for per-video classification on KTH dataset using LDCRFs at window size = 4.

Action	Walk	Run	Jog	Box	Wave	Clap
Walk	1.00	0.00	0.00	0.00	0.00	0.00
Run	0.00	0.98	0.02	0.00	0.00	0.00
Jog	0.00	0.02	0.98	0.00	0.00	0.00
Box	0.00	0.00	0.00	0.98	0.00	0.02
Wave	0.00	0.00	0.00	0.00	1.00	0.00
Clap	0.00	0.00	0.00	0.03	0.00	0.97

To assess the efficiency of the proposed method, the obtained results have been compared with those of other previously published studies in the literature, as shown in Table 3. From this comparison, it turns out that our approach using LDCRfs performs competitively with other state-of-the-art approaches, and its results compared favorably with previously published results. Notably, all the methods that we compared our method with have used similar experimental setups. Thus, the comparison is meaningful.

Table 3: Comparison with the state-of-the-art on KTH dataset Uing LDCRFs at window size=4.

Method	Recognition rate
Our Method	98.50%
Ke *et al.* [8]	63.00%
Liu and Shah[9]	92.80%
Jhuang *et al.* [10]	91.70%
Wang and Mori [11]	92.50%
Rapantzikos *et al.* [12]	88.30%
Dollár *et al.* [13]	81.20%

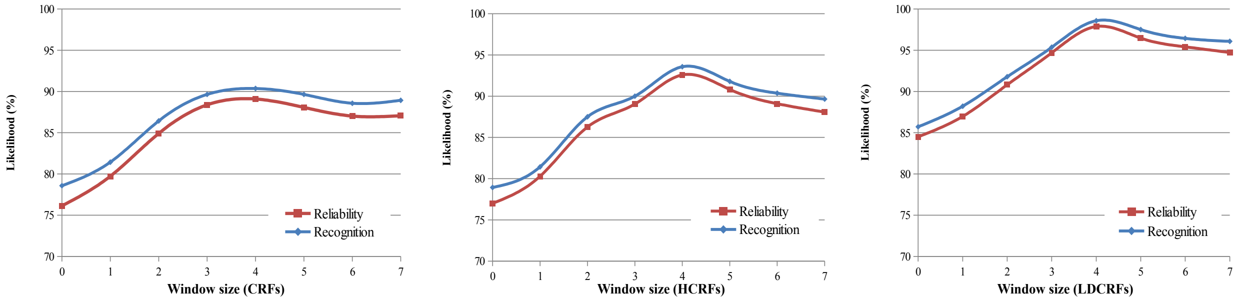

Fig. 9: The recognition and reliability accuracy of KTH dataset using CRFs, HCRFs and LDCRFs relative to sliding window size ranging from 0 to 7.

4.3 Evaluation

The sample test data is entirely different from the training data and is tested on *Intel(R) Core(TM) i7 CPU 3.4 GHz PC with 4 GB of RAM*. The time complexity of the CRFs matching model presented throughout this work is proportional to the number of cells, which are visited by dynamic programming method. CRFs take $O(PL^2)$ where L is either six labels for human actions, and P is the number of input feature vectors at every time instance. The space complexity for the matching algorithm is similar to the time complexity if the proposed approach is running in offline modes. The following algorithm summarizes the matching process of CRFs models for a given observation sequence.

```
Input: An observation sequence $x$, $T$ represents the length
       of $x$ and the number of labels is $L$
Output: Probability of label sequence $y$ given CRFs
        parameters: $p(y|x,\theta)$

$i$ = 1, initialize $Z$
while $i \leq T$ do
    for $j$ = 1 to $L$ do
        for $k$ = 1 to $L$ do
            $M_i(y_j, y_k) =$
            $\exp\left(\sum_f \lambda_f t_f(y_j, y_k, x, i) + \sum_g \mu_g s_g(y_k, x, i)\right)$
        end
    end
    $Z = Z \times M_i$ % $Z$ is a normalization factor
    $q^* = M_i(y_{i-1}, y_i|x)$ % $q^*$ is a product of all matrices $M$
    $i = i + 1$
end
$p(y|x,\theta) = \frac{1}{Z} \times q^*$
```

Algorithm 1: Matching CRFs model

Although, CRFs, HCRFs and LDCRFs show an extremely strong performance, they are very expensive in terms of training costs. Fig. 10, summarize the time costs of the used discriminative models on IIKT and HKT datasets, and further indicating the cost proportional to the window size. Additionally, illustrate that LDCRFs models that allow the representation of hidden states are usually more expensive than their standard counterparts. The high recognition rate achieved is due to the following reasons; 1) A high segmentation accuracy of the foreground is achieved. 2) A set of feature candidates who optimally discriminate among the input actions is elected. 3) A carefully experimental based selection of initialization parameters for training process. 4) LDCRFs classification technique has the ability to alleviate spatio-temporal variabilities.

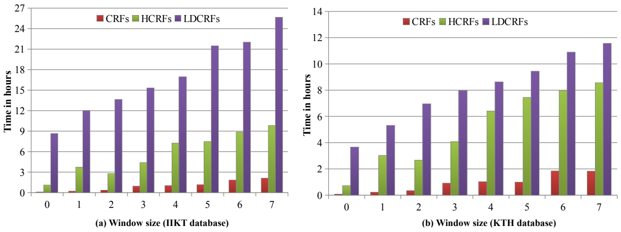

Fig. 10: Effects of different window size on the training cost in terms of time.

5 Conclusion

In this paper, we proposed an approach to investigate humane action recognition based on an affine-invariant shape descriptor like SCLF, in addition to mass center and optical flow motion features. The extracted features from 3D spatio-temporal action volumes are employed with varying windows size for the discriminative models as CRFs, HCRFs and LDCRFs for recognition. As a result, the reliability of human activity is improved initially as the window size increase, but degrades as the window size increase further. Our experiments on standard benchmark action KTH and IIKT datasets show that the proposed LDCRFs approach is more robust and yields promising results when comparing favorably with those previously reported without sacrificing real-time performance. The future research will address the empirical validation of the approach on more realistic datasets presenting many technical challenges in data handling, such as object occlusion and significant background clutter.

Acknowledgement

The authors are grateful to the anonymous referee for a careful checking of the details and for helpful comments that improved this paper.

References

[1] R. Poppe, Journal of Image and Vision Computing **28**, 976–990 (2010).

[2] S. Sadek, A. AI-Hamadi, G. Krell, B. Michaelis, Scientific World Journal: ISRN Machine Vision 2013, 1–7 (2013).

[3] A. A. Efros, A. C. Berg, G. Mori, J. Malik, IEEE Conf. on Computer Vision, 726–733 (2003).

[4] A. Bobick, J. Davis, IEEE Trans. on Pattern Analysis and Machine Intelligence **23**, 257–267 (2001).

[5] L. Zelnik-Manor, M. Irani, IEEE on Computer Vision and Pattern Recognition, 1–8 (2001).

[6] E. Shechtman, M. Irani, IEEE on Computer Vision and Pattern Recognition, 405–412 (2005).

[7] M. Ahmad, S.-W. Lee, Journal of Pattern Recognition **41**, 2237–2252 (2008).

[8] Y. Ke, R. Sukthankar, M. Hebert, IEEE International Conference on Computer Vision, 166–173 (2005).

[9] J. Liu, M. Shah, IEEE Conf. on Computer Vision and Pattern Recognition, 1–8 (2008).

[10] H. Jhuang, T. Serre, L. Wolf, T. Poggio, IEEE International Conf. on Computer Vision, 1–8 (2007).

[11] Y. Wang, G. Mori, IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 872–879 (2009).

[12] K. Rapantzikos, Y. Avrithis, S. Kollias, IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 1454–1461 (2009).

[13] P. Dollár, V. Rabaud, G. Cottrell, S. Belongie, IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 65–72 (2005).

[14] A. Fathi, G. Mori, IEEE Conf. on Computer Vision and Pattern Recognition, 1–8 (2008).

[15] M. Bregonzio, S. Gong, T. Xiang, IEEE Conf. on Computer Vision and Pattern Recognition, 1948–1955 (2009).

[16] Z. Zhang, Y. Hu, S. Chan, L. T. Chia, European Conference on Computer Vision, 817–829 (2008).

[17] J. C. Niebles, H. Wang, L. Fei-Fei, Journal of Computer Vision **79**, 299–318 (2008).

[18] C. Schueldt, I. Laptev, B. Caputo, International Conference on Pattern Recognition **3**, 32–36 (2004).

[19] L. Gorelick, M. Blank, E. Shechtman, M. Irani, R. Basri,IEEE Trans. on Pattern Analysis and Machine Intelligence **29**, 2247–2253 (2007).

[20] C. Sminchisescu, A. Kananujia, D. Metaxas, Journal of CVIU **104**, 210–220 (2006).

[21] S. Sadek, A. Al-Hamadi, B. Michaelis, U. Sayed, IEEE Conf. on ICIP, 765–7680 (2012).

[22] C. Shi, B. Wang, Advances in Image and Video Technology, First Pacific Rim Symposium, 403–410 (2006).

[23] J. Lafferty, A. McCallum, F. Pereira, Conf. on ICML, 282–289 (2001).

[24] H. Yang, S. Sclaroff, S. Lee, IEEE Trans. on PAMI **31**, 1264–1277 (2009).

[25] A. Gunawardana, M. Mahajan, A. Acero, J. C.Platt, Proceeding of European Conf. on Speech Communication and technology, 1117–1120 (2005).

[26] A. Quattoni, S. Wang, L. P. Morency, M. Collins, T. Darrell, IEEE Trans. on PAMI **29**, 1848–1852 (2007).

[27] L. P. Morency, A. Quattoni, T. Darrell, IEEE conf. on CVPR, 1–8 (2007).

[28] S. J. McKenna, Y. Raja, S. Gong, Journal of Image and Vision Computing **17**, 225–231 (1999).

[29] M. Elmezain, , International Journal of Engineering Science and Innovative Technology (IJESIT) **2**, 438–445 (2013).

[30] K. Liu, Q. Du, H. Yang, B. Ma, EURASIP Journal on Advances in Signal Processing, 1–6 (2010).

[31] A. McCallum, Conf. on Uncertainty in AI, 403–410 (2003).

[32] H. M. Wallach, , Tech. Report Ms-CIS-04-21, University of Pennsylvania.

Mahmoud Elmezain was born in Egypt. Between 1997 and 2004 he worked as Demonstrator in Dept. of Statistic and Computer Science, Tanta University, Egypt. He received his Masters Degree in Computer Science from Helwan University, Egypt in 2004. He received PhD Degree in Computer Science from Institute for Electronics, Signal Processing and Communications at Otto-von-Guericke-University of Magdeburg, Germany. His work focuses on image processing, pattern recognition, human-computer interaction and action recognition..

Essam O. Abdel-Rahman was born in Egypt. Between 1998 and 2003 he worked as Demonstrator in Dept. of Mathematics, Faculty of scince, AL-Azhar University, Assuit branch, Assuit, Egypt. He received his Masters Degree in Numerical Analysis from Assuit University, Assuit, Egypt in 2003. He received PhD Degree in Computer Science for Institute of Computer Science II at Bonn University, Bonn, Germany. His work focuses on Study of Bifurcation Autonomous Parametric Dynamical System, Practical Computer Science, Image processing, Computer Graphic, and Algorithmic Contribution.