Skeleton-based Human Activity Recognition for Video Surveillance

Ahmed Taha [1], Hala H. Zayed [1], M. E. Khalifa [2] and El-Sayed M. El-Horbaty [3]
[1] Computer Science Dept., Faculty of Computers & Informatics, Benha University.
[2] Basic Science Dept., Faculty of Computer & Information Sciences, Ain Shams University.
[3] Computer Science Dept., Faculty of Computer & Information Sciences, Ain Shams University.
{ahmed.taha, hala.zayed}@fci.bu.edu.eg, {esskhalifa, shorbaty}@cis.asu.edu.eg

Abstract— Recognizing human activity is one of the important areas of computer vision research today. It plays a vital role in constructing intelligent surveillance systems. Despite the efforts in the past decades, recognizing human activities from videos is still a challenging task. Human activity may have different forms ranging from simple actions to complex activities. Recently released depth cameras provide effective estimation of 3D positions of skeletal joints in temporal sequences of depth maps. In this paper, a system for human activity recognition is proposed. We have considered the task of obtaining a descriptive labeling of the activities being performed through labeling human sub-activities. The activities we consider happen over a long period, and comprise several sub-activities performed in a sequence. The proposed activity descriptor makes the activity recognition problem viewed as a sequence classification problem. The proposed system employs Hidden Markov Models (HMMs) to recognize human activities. The system is evaluated on two benchmark datasets for daily living activity recognition. Experiment results demonstrate that the proposed system outperforms the state-of-the-art methods.

Index Terms— Activity Recognition, Behavior Analysis, Depth Images, HMM, MSVM, RGB-D, Video Surveillance

1 Introduction

RECENTLY, surveillance systems have been very useful in all public and private sectors. The increasing global security concerns have resulted in more research in samrt surveillance. It has a wide range of applications including effective monitoring of public places such as airports, railway stations, shopping malls, crowded sports arenas, military installations, etc., or for use in smart healthcare facilities such as daily activity monitoring and fall detection in old people's homes [1]. Conseqently, there is an urgent need to analyze human behaviors in video surveillance systems automatically. Human monitoring of surveillance video is a very labor-intensive task. Detecting multiple activities in real-time video is difficult in manual analysis. Thus, the intelligent video surveillance system is emerged. Automatic behavior analysis involves the analysis and the recognition of motion patterns to produce a high-level description of actions and interactions among objects [2]. Despite significant research efforts over the past few decades, action recognition remains a highly challenging problem. The difficulties of action recognition come from several aspects [3, 4]. Firstly, human motions are represented in a very high dimensional space. Moreover, interactions among different subjects complicate searching in this space. Secondly, performing similar or identical activities by different subjects exhibit substantial variations. Thirdly, visual data from traditional video cameras can only capture projective information of the real world, and are sensitive to lighting conditions.

The problem of behavior analysis is addressed under different terms. In the literature, action recognition and activity recognition are the most common used terms [2, 5]. The term action is often confused with the term activity. Action usually refers to a sequence of primitive movements carried out by a single object, that is, an atomic movement that can be described at the limb level [5], such as a walking step. However, activity contains a number of sequential actions. i.e., dancing activity consists of successive repetitions of several actions, e.g. walking, jumping, waving hand, etc. Actions can be placed on a lower level than activities. Approaches for recognizing activities are often hierarchical in nature. They use previously recognized actions as their input. Different approaches are used to recognize low-level actions [6]. Some approaches use every single frame (2D templates, 3D object models), while others look at the entire video (spatio-temporal filtering, sub-volume matching). These techniques extract features and match them to a template in order to recognize an action. Other techniques, such as hidden Markov models (HMMs), estimate a model on the temporal dynamics of an action. The model parameters are learned from training data.

From a representation simplicity viewpoint, low-level features (such as pixels) and spatiotemporal features have achieved promising performances on some of the benchmark datasets. Actually, low-level features benefit from the fact that they are generally easy to extract. However, they are unable to handle the temporal structure of the action/ behavior. Thus, there is a need for a higher-level analysis to construct a suitable temporal model. Mid and high-level representations can bridge this gap. One of the most common methods for representing human action is the use of human's skeletal information. In the past, extracting accurate skeletal information from video streams was very difficult and unreliable, especially for arbitrary human poses. In con-

trast, motion capture systems could provide very accurate skeletal information of human actions based on active or passive markers positioned on the body [7]. However, the data acquisition was limited to controlled indoor environments. Hence, skeletal-based recognition methods became less popular over the years as compared to the image feature-based recognition methods [7]. The latter methods extract spatiotemporal interest points from video images and the recognition is based on learned statistics on large datasets. Lately, new technologies help to enhance the monitoring process creating systems that are more powerful in detecting dangerous situations. With the release of several low-cost 3D capturing systems, such as the Microsoft Kinect, real time 3D data acquisition and skeleton extraction have become much easier and more practical for action recognition, thus restoring interest in the skeleton-based action recognition.

In this paper, a system for human activity recognition is proposed. Actually, we extend our previous work presented in [8] by focusing on recognizing complex activities as a sequence of basic actions. The proposed system presents a human activity descriptor based on the human's skeletal information extracted from Microsoft Kinect. This representation of the human activity is invariant to the scale of the subjects/objects and the orientation to the camera, while it maintains the correlation among different body parts. Hidden Markov Models (HMMs) are employed to recognize human activities. For each activity class, a HMM is learned. In the classification step, an unknown activity descriptor is aligned with the HMM in each class. An unknown sequence will be classified into the class, which has the highest alignment score.
The remainder of this paper is organized as follows: Section 2 gives an overview about RGB-D sensor and depth images pointing out their advantages over the intensity images. In Section 3, we briefly review some related work in human activity recognition. Section 4 then presents the proposed system. The performance analysis of the proposed system is empirically evaluated in Section 5. Finally, we conclude in Section 6.

2 RGB-D Sensor

The RGB-D sensor (such as Microsoft Kinect) is a motion-capture device that provides the 3D location and skeleton posture of the human body [9]. The Kinect sensor produces a new type of data, RGB-D data, which is an improvement on RGB images for human behavior recognition research. Its name is a combination of kinetic and connects [10]. It was initially used as an input device by Microsoft for the Xbox game console. All user movements are captured and reflected on-screen. It enables the user to interact and control software on the Xbox 360 with gestures recognition and voice recognition. The Kinect's output is a multi-modal signal, which gives RGB videos, depth sequences and skeleton information simultaneously. Recently, the computer vision community discovered that the depth sensing technology of Kinect could be extended far beyond gaming and at a much lower cost than traditional 3D cameras (such as stereo cameras and Time-Of-Flight cameras) [11]. Prior to the Kinect, a 3D laser scanner was the primary device to capture accurate 3D depth data of a scene. However, the huge volume and high price of the laser scanner restrict its usage in many applications [4]. Another alternative is the use of a stereo vision system consisting of two cameras to get 3D information. Nevertheless, the resolution of the cameras, the calibration of the system and the required heavy computations increase the complexity of the system and greatly affect the accuracy of the 3D depth data [4].

Figure 1 shows the Kinect sensor and the RGB-D data captured including both RGB color image and depth image. A time of flight camera is implicitly embedded in the Kinect. It measures the distance of any given point from the sensor using the time taken by near-IR light to reflect from the object. In addition to it, an IR grid is projected across the scene to obtain deformation information of the grid to model surface curvature. A depth image (or depth map) is an image that contains information relating to the distance of the surfaces of scene objects from a viewpoint [12]. Pixels in a depth image indicate calibrated depth in the scene, rather than a measure of intensity or color. The device is actually composed of multiple sensors. In the middle, it has a RGB camera allowing a resolution up to 1280×960 at 12 images per second [10]. The usual used resolution is 640×480 pixels at 30 images per second maximum for colored video stream as the depth camera has a maximum resolution of 640×480 at 30 frames per second. A little away on the left of the device, It has the IR light (projector). It projects multiple dots, which allow the final camera on the right side, the CMOS depth camera, to compute a 3D environment. The device is mounted with a motorized tilt to adjust the vertical angle.

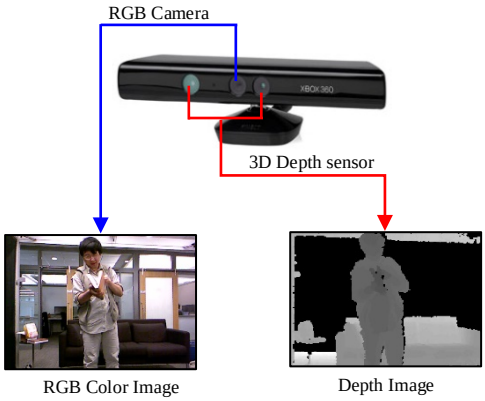


Fig. 1. RGB-D data captured by Kinect

One of the major components of the Kinect sensor is its ability to infer human motion by extracting human silhouettes in skeletal structures. It extracts the skeletal joints of a human body as 3D points using the Microsoft SDK. It provides a skeleton model with 20 joints as shown in Figure 2. The complementary nature of the depth and visual RGB information provided by Kinect initiates new solutions for classical problems in computer vision. The availability of depth information allows researchers to implement simpler identification procedures to detect human subjects. The advantages of this technology, with respect to classical video-based ones, are [13]:

- Being less sensitive to variations in light intensity and texture changes;
- Providing 3D information by a single camera, while a stereoscopic system is necessary in the RGB domain to achieve the same goal;
- Maintaining privacy, it is not possible to recognize the facial details of the people captured by the depth camera. This feature helps to keep identity confidential.

ID	Joint
1	HipCenter
2	Spine
3	ShoulderCenter
4	Head
5	ShoulderLeft
6	ElbowLeft
7	WristLeft
8	HandLeft
9	ShoulderRight
10	ElbowRight
11	WristRight
12	HandRight
13	HipLeft
14	KneeLeft
15	AnkleLeft
16	FootLeft
17	HipRight
18	KneeRight
19	AnkleRight
20	FootRight

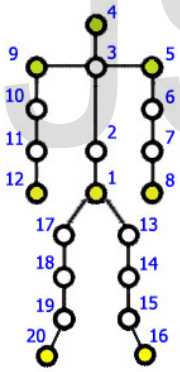


Fig. 2. Skeleton joints tracked by the Kinect Sensor using Microsoft SDK

3 Literature Review

During the past few years, a rich palette of diverse ideas has been proposed on the problem of recognition of human activities by employing different types of visual information. However, the problem is still open and provides a big challenge to the researchers and more rigorous research is needed to come around it. An overview of the various action recognition methods and available well-known action datasets are provided in [14, 15]. Most previous research in action recognition was based on color or greyscale intensity images. These images are obtained from traditional RGB cameras, where the value of each pixel represents the intensity of incoming light. It contains rich texture and color information, which is very useful for image processing, however it is very sensitive to illumination changes.

Recently, there have been vision technologies that can capture distance information from the real world, which cannot be obtained directly from an intensity image. These images are obtained from depth cameras, where the value of each pixel represents the calibrated distance between camera and scene. An advantage of using these sensors is that they give depth at every pixel so the shape of the object can be measured. When using depth images, computer vision tasks like background subtraction and contour detection become easier. Actually, there are many attractive progresses and improves have been done with the use of depth information.

Based on the above, there are two main approaches for human behavior recognition: RGB video-based approach [15] and depth map-based approach [3, 4]. In this section, we focus only on reviewing the state-of-the-art techniques that investigate the applicability and benefit of depth sensors for action recognition. In [16] Sung et al. present a two-layered Maximum Entropy Markov Model (MEMM). It models different properties of the human activities, including their hierarchical nature, the transitions between sub-activities over time, and the relation between sub-activities and different types of features. They use a RGBD sensor (Microsoft Kinect) as the input sensor, and compute a set of features based on human pose and motion, as well as based on image and point-cloud information. During inference, their algorithm exploits the hierarchical nature of human activities to determine the best MEMM graph structure. It infers the two-layered graph structure using a dynamic programming approach.

Also, Ni et al. [17] propose a complex activity recognition and localization framework that fuses information from both grayscale and depth image channels at multiple levels of the video processing pipeline. In the individual visual feature detection level, depth-based filters are applied to the detected human/object rectangles to remove false detections. In the next level of interaction modeling, 3-D spatial and temporal contexts among human subjects or objects are extracted by integrating information from both grayscale and depth images. Depth information is utilized to distinguish different types of indoor scenes. Finally, a latent structural model is developed to integrate the information from multiple levels of video processing for an activity detection.

Gupta et al. [18] prsent a method to classify human activities by leveraging on the cues available from depth images alone. They propose a descriptor, which couples depth and spatial information of the segmented body to

describe a human pose. Unique poses (codewords) are then identified by a spatial-based clustering step. Given a video sequence of depth images, they segment humans from the depth images and represent these segmented bodies as a sequence of codewords. They exploit unique poses of an activity and the temporal ordering of these poses to learn subsequences of codewords, which are strongly discriminative for the activity. Each discriminative subsequence acts as a classifier and they learn a boosted ensemble of discriminative subsequences to assign a confidence score for the activity label of the test sequence.

Morover, Koppula et al. [19] consider the task of jointly labeling human sub-activities and object affordances in order to obtain a descriptive labeling of the activities being performed in the RGB-D videos. They jointly model the human activities and object affordances as a Markov Random Field (MRF) where the nodes represent objects and sub-activities, and the edges represent the relationships between object affordances, their relations with sub-activities, and their evolution over time. The parameters of the model are learned using a structural SVM formulation. Their model also incorporates the temporal segmentation problem by computing multiple segmentations and considering labeling over these segmentations as latent variables.

In [20], Koppula and Saxena extended the work presented in [19] by detecting the past human activities as well as anticipating the future human activities using object affordances. In the detection process, they present a method to first obtain potential graph structures that are close to the ground-truth ones by approximating the graph with only additive features. Starting with this graph structure, they then design moves to obtain several other likely graph structures to be used in the anticipating process.

In [21], Hu et al. present a latent discriminative model for human activity recognition. The parameters of the graphical model are learned with the Structured-Support Vector Machine (Structured-SVM). A data-driven approach is used to initialize the latent variables, thereby no hand labeling for the latent states is required. By making the observation and state nodes fully connected, the model do not require any conditional independence assumption between latent variables and the observations.

4 PROPOSED SYSTEM

The proposed method focuses on obtaining a descriptive labeling of the complex human activities that take place over different time scales and consist of a sequence of sub-activities (actions). In fact, human activity recognition is a challenging task since it needs to face with numerous varieties. First, the variation in the length of an action where different individuals perform actions at diverse rate. Second is differences in the characteristics of the human body such as body shape, height, weight fitting, etc. Third is the ambiguity caused by the similarity of some activities, which represents a great challenge for any recognition system. Moreover, environment settings and video quality should be considered. For example, dynamic backgrounds and cluttered environments are always difficult to handle in any video processing application. Other factors such as lighting condition, camera viewpoint, and camera motion should also be addressed properly.

In fact, our previous work in [8] focuses on recognizing actions that span short time periods. However, in this paper, the proposed system extends that work by performing a high-level activity recognition. These activities take place over a long period and consist of a sequence of sub-activities. The proposed system employs the human action representation presented in [8] to recognize complex activities. This representation is characterized by its low dimensionality and its invariance to the scale of the subjects/objects and the orientation to the camera, while it maintains the correlation among different body parts. It is based on the human's skeletal information extracted from depth images. The basic idea of the proposed system depends on the fact that each activity consists of a sequence of sub-activities (actions) that change over the course of performing the activity. The proposed system recognizes these actions independently. Then, an activity descriptor is constructed from these actions as an ordered sequence. Initially, the descriptor is empty. Then, every detected action is added in order to the sequence. Later, trained Hidden Markov Models (HMMs) are used for recognizing unknown activities.

Figure 3 shows the block diagram of the proposed system. First, the system starts with identifying the skeleton joints coordinates for each detected object in the video sequence. Actually, the Kinect camera tracks 20 body joints for each object in the scene. The position of the skeleton joints are provided as Cartesian coordinates (X, Y, Z) with respect to a coordinate system centered at the Kinect. The positive Y axis points up, the positive Z axis points where the Kinect is pointing, and the positive X axis is to the left as shown in Figure 4.

Second, the proposed system constructs the feature vector for each detected skeleton in the scene. Ideally, a subject should be straight in front of Kinect camera (Figure 5.a) but this is not always the case. The subject can be at any angle from Kinect (Figure 5.b) and at any distance. To overcome this issue, the proposed system rotates all the skeleton points around Y-axis in a counterclockwise

direction with an angle α in order to make the subject straight in front of depth camera. Hence, rotation invariance is achieved. This angle is defined as the angle between the line connecting both shoulders and the positive direction of X-axis of Kinect coordinates system (Figure 5.b). Initially, the angle α is estimated using the coordinates of two joints: shoulder left (x_L, y_L, z_L) and shoulder right(x_R, y_R, z_R) through the following equation:

$$\propto = \tan^{-1}\left(\frac{z_R - z_L}{x_R - x_L}\right)$$

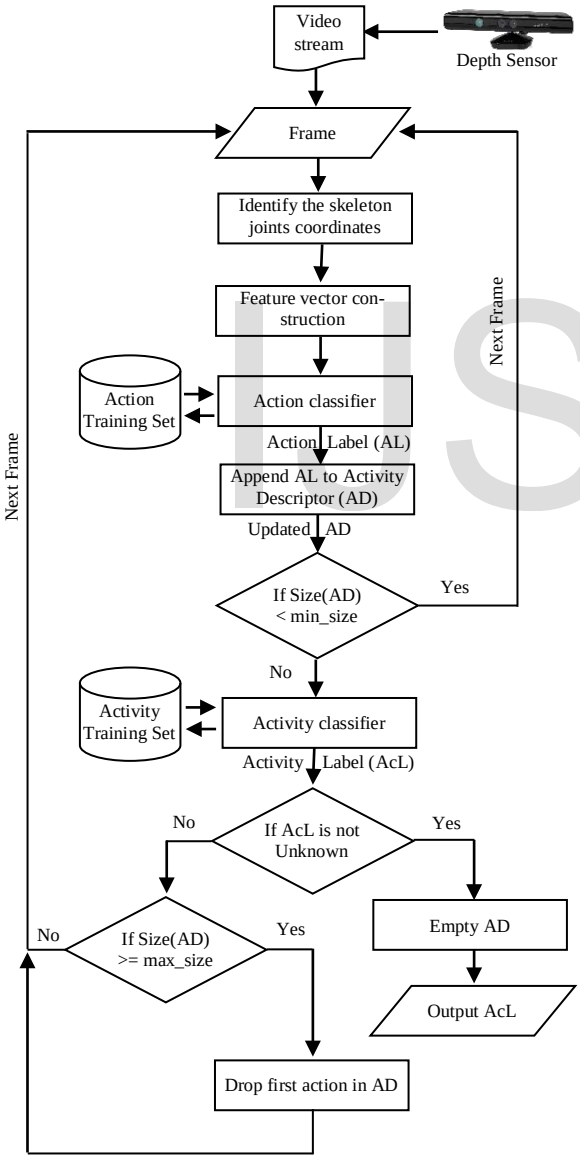


Fig. 3. The block diagram of the proposed system

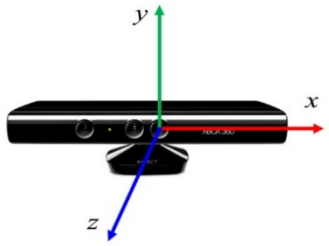
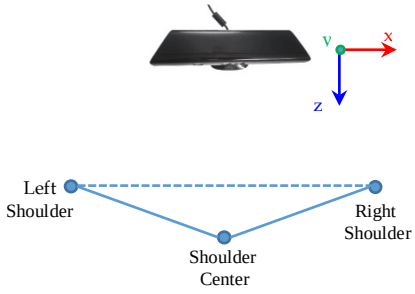
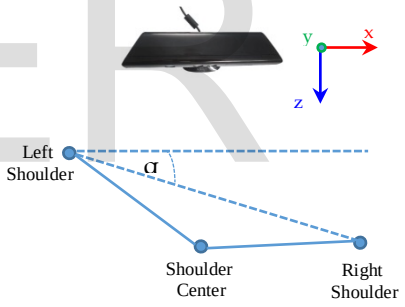


Fig. 4. Kinect Cartesian coordinate system



(a) The subject facing the depth camera



(b) The subject at an angle with respect to the depth camera

Fig. 5. Rotation of the skeleton with respect to the Kinect

Then a counterclockwise rotation about Y-axis is applied to all skeleton joints with an angle α. For each skeleton joint i with coordinates(x_i, y_i, z_i), the rotated coordinates (x_i', y_i', z_i') are calculated with the following transformation:

$$\begin{bmatrix} x_i' \\ y_i' \\ z_i' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \propto & 0 & \sin \propto & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \propto & 0 & \cos \propto & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}$$

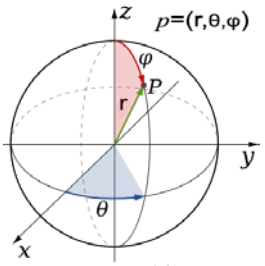
Moreover, varying the object distance from Kinect makes the action recognition more sophisticated. Therefore, it is necessary to shift the origin of the coordinates from Kinect to a point in the object body to remove dependence

on camera position. This means joints coordinates should be translated to another coordinate system where its origin is a point in the human body rather than the Kinect camera. By this way, the distance factor between the object and Kinect is neutralized. This permits the coordinates to be expressed invariantly to translation and rotation of the body with respect to the camera reference system. In our proposed system, we use the shoulder center joint as the origin of the new system (see Figure 2). Assume that shoulder center joint coordinates are (x, y, z). Hence for each skeleton joint i with coordinates (x_i, y_i, z_i), the translated coordinates (x_i', y_i', z_i') are calculated with the following equation:

$$(x_i', y_i', z_i') = (x_i - x, y_i - y, z_i - z)$$

Moreover, the individual variations of people in terms of posture, height and dimensions have a huge impact on the performance of the action recognition system. This is because X, Y and Z coordinates of joints of every object doing the same action might be different. Therefore, it is necessary to normalize the data to increase accuracy of action recognition. To simplify the normalization process, the joints coordinates are converted from Cartesian coordinate system to spherical coordinate system. The spherical coordinate system is a three dimensional space system with three components: the distance of the point from the origin (radial distance r), the polar angle (ϕ), and the azimuth angle (θ) as shown in Figure 6. When normalizing a point in Cartesian coordinates, all the components X, Y and Z are changed. However when normalizing a point in the spherical coordinates, only radial distance r will equal to one while both polar angle (ϕ) and azimuth angle (θ) will remain constant..

Feature vectors provide a set of characteristics that represent the action to be recognized. However, it may include irrelevant or redundant information which could complicate the classification. Reducing the feature vector size has an important impact on the processing time since the recognition is performed faster. Concerning the skeletal data obtained with depth sensor devices, it can be seen that some joints are more important than others if action recognition is targeted. Several joints in the torso (the skeleton part identified by a dashed line in Figure 7) do not show an independent motion along with the whole body. Hence, in our proposed system, seven joints coordinates of the human skeleton are discarded from the feature vector. These joints are shown as solid circles in Figure 7: shoulder right, shoulder center, shoulder left, spine, hip center, hip right, and hip left (from left-to-right and from top-to-bottom respectively). This dimensionality reduction of the feature vector improves the classification performance. Since the joints coordinates are normalized, radial distance r can be ignored in our feature vector. Thus, the feature vector will consist of 13 pairs of (ϕ , θ) for each detected object in the scene. This means it has only 26 components which is a reduced feature vector than what is reported in the state-of-the-art methods [16-21]. A low-dimensional representation means less computational effort.



$$r = \sqrt{x^2 + y^2, z^2}\,, \quad \theta = \cos^{-1}\left(\frac{z}{r}\right), \quad \varphi = \tan^{-1}\left(\frac{y}{x}\right)$$

Fig. 6. Spherical coordinates (r, θ, φ): radial distance r, azimuthal angle θ, and polar angle φ

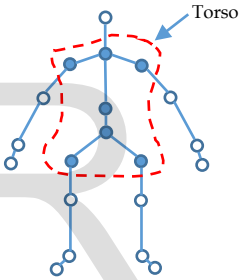


Fig. 7. Torso skeleton joints discarded from the feature vector

After a feature vector is constructed, a classification step is needed to recognize different actions. The feature vector of the unknown action is used as input to the classifier whose objective is to accurately identify which action class is best matched against the input. In our proposed system, a Multi-class Support Vector Machine (MSVM) [22-24] is employed to perform action classification. The MSVM used is based on One-Against-All (OAA) classification approach [23] where there is one binary SVM for each class to separate members of that class from members of other classes. A data point would be classified under a certain class if and only if that class's SVM accepted it and all other classes' SVMs rejected it. A training step is needed to summarize the similarity within (and dissimilarity in-between) the training samples of different action classes. With action models learned, a new action instance can be recognized as one of the learned classes.

Once an action is recognized, it is a candidate to be a part of a more complex activity. This is because a human

activity is actually a series of human actions. In order to recognize this activity, the proposed system constructs and maintains an activity descriptor. It is simply an ordered list of the detected actions and it satisfies two criteria. First, adjacent actions in the activity descriptor are not allowed to be the same. However, the activity descriptor may contain the same action more than one time but not adjacent. Second, the activity descriptor is variable length with a special notion of order since not all activities consist of the same number of actions. However, a minimum and a maximum size of the descriptor is initially predetermined from the training set. Initially, the activity descriptor is an empty set and it is updated each time either an action or an activity is recognized.

Considering the nature of the proposed activity descriptor, the problem of recognizing activities can be formulated as a sequence classification problem. Given L as a set of class labels, the task of sequence classification is to learn a sequence classifier C, which is a function mapping of a sequence s to a class label $l \in L$, written as, $C : s \rightarrow l; l \in L$. In the proposed system, HMMs are employed for performing action recognition, due to their suitability for modeling pattern recognition problems that exhibit an inherent temporality. HMMs are one of the most popular generative models used for classification. It is a doubly stochastic process [25]. The underlying stochastic process is not observable but can be observed through another set of stochastic processes that produce the sequence of observed symbols [25]. The underlying hidden stochastic process is a first-order Markov process; that is, each hidden state depends only on the previous hidden state. Moreover, in the observed stochastic process, each observed measurement (symbol) depends only on the current hidden state. The use of HMMs includes two stages: learning and recognition. In the learning stage, the data are used to optimize the parameters of the HMM of each activity (class). That is, it involves developing a model for all of the activities that we want to recognize. In the recognition stage, the HMM of each class computes the probability of generating a test sequence, and the model which has the maximum probability is chosen.

Back to Figure 3, when an action is recognized, the action is appended to the activity descriptor provided it does not match the last action in the descriptor. If the descriptor size is less than the minimum size, the proposed system will proceed to the next frame to detect more actions to be added to the descriptor. Otherwise, when the descriptor reaches the minimum size, it is a candidate to be an activity. At this point, the activity descriptor is checked against all the trained HMMs to calculate the likelihood and the one having highest probability is chosen. Thus, to test an activity descriptor sequence AD, the HMMs act as:

$$AcL = \arg \max_{i=1,2,\ldots,N} \{P(AD|H_i)\}$$

where the activity label (AcL) is based on the probability of the activity descriptor (AD) on corresponding trained activity HMM H_i. When an activity is recognized, the proposed system resets the descriptor. It becomes empty again and ready for receiving more actions of the next activity. However, if the activity is not recognized, so the actions in the descriptor are not sufficient to recognize the activity. In this case, the descriptor size is checked against reaching to the maximum size. If so, the first action in the descriptor is dropped leaving the empty space for adding one more action. Otherwise. The proposed system proceeds to the next frame to recognize next actions. In this way, the system is able to recognize different human activities.

In surveillance applications, activities of interest usually occur rarely. Suspicious activity may take many different forms. It can be classified as either being normal activity but appeared in a different context or being abnormal and unexpected activity occureed rarely. Figure 8 shows a simple procedure used by our proposed system to classify the detected activity as either suspicious or not. It is invoked after the recognition system outputs its results.

For the first type, it should be noted that the definition of unusual activities is rather subjective. What is considered suspicious on one place may be normal activity on another place. For example, the "Runing" activity in a bank hall is considered suspicious while it is normal activity in a stadium or in a park. Also, the "Loitering" activity (random walk) around schools, parking lots or secluded areas is considered suspicious while it is normal activity if an individual is waiting for a bus at a terminal or if a person is going for a walk in a park. Even "walking" activity can be considered suspicious if it was a walk in a restricted area. This type of activities can be easily recognized by our proposed system since they are actually well defined activities. A list of suspicious activities is initially predetermined and the system can be trained efficiently to recognize them.

For the second type of suspicious activities, unusual (or abnormal) activities, it is difficult to collect sufficient training data for supervised learning. In this case, many unusual activity detection algorithms, which require large numbers of training data, become unsuitable. Moreover, clusters for these activities may not be representative enough to predict future unusual ones. Our proposed system addresses the lack-of-training-data problem of unusual activities by classifying any unrecognized detected activity as a candidate to be suspicious one. Actually, this activity seldom occurs or has not been

observed before, i.e. having low statistical representation in the dataset. In this case, the activity needs further examination by a human operator.

Suspicious Activity Detection Procedure:

Inputs: Activity label (AcL) [output of the activity classifier]

Output: Suspicious Activity, Un-suspicious Activity

Steps:

1. If AcL is "Unknown"
2. Return "Suspicious Activity"
3. Else if AcL exists in Suspicious Activity List
4. Return "Suspicious Activity"

Else

5. Return "Un-suspicious Activity"

Fig. 8. The pseudo code of the Suspicious Activity Detection Procedure

5 Experimental Results

In order to evaluate the performance of the proposed system, two benchmark datasets are used: Cornell CAD-60 [26] and Cornell CAD-120 [19]. Both datasets have five different environments of a regular household including bathroom, bedroom, kitchen, living room, and office. The datasets consider three to four common activities identified in each location with about 45 seconds of data for each activity and each person. The CAD-120 extends the CAD-60 by separating high-level human-object interactions (e.g. taking medicine, cleaning objects, microwaving food) and sub-activities like punching, reaching, and drinking.

The Cornell CAD-60 dataset (available at: http://pr.cs.cornell.edu/humanactivities) has 60 RGB-D videos of four different subjects (two males and two females). They perform 12 high-level activity classes including: rinsing mouth, brushing teeth, wearing contact lens, talking on the phone, drinking water, opening pill container, cooking (chopping), cooking (stirring), talking on couch, relaxing on couch, writing on whiteboard, and working on computer. Figure 9 shows some example frames of Cornell CAD-60 dataset. Note that some of the activity classes in CAD-60 dataset contain only one sub-activity (e.g. working on a computer, cooking (stirring), etc.) and do not contain object interactions (e.g. talking on couch, relaxing on couch).

The second dataset used in the experiments is the Cornell CAD-120 dataset (available at: http://pr.cs.cornell.edu/humanactivities). It contains 120 activity sequences of ten different highlevel activities and ten different sub-activities. Four different subjects (two males and two females) perform each activity three times. They performed the activities through a long sequence of sub-activities, which varied from subject to subject significantly in terms of length of the sub-activities and in the way they executed the task. The dataset contains a total of 61,585 RGB-D video frames. The ten high-level activities include: making cereal, taking medicine, stacking objects, unstacking objects, microwaving food, picking objects, cleaning objects, taking food, arranging objects, and having a meal, while sub-activity labels include: reaching, moving, pouring, eating, drinking, opening, placing, closing, scrubbing, and null. Figure 10 shows some example frames of these sub-activity labels. Note that the subjects perform the high-level activities multiple times with different objects. For example, the stacking and unstacking activities were performed with pizza boxes, plates and bowls. Table 1 specifies the set of sub-activities involved in each high-level activity. For example, the "making cereal" activity consists of the following sub-activities: 1) placing bowl on table, 2) pouring cereal, 3) pouring milk. for "microwaving food" activity, it consists of: 1) opening microwave door, 2) placing food inside, 3) closing microwave door. Note that some activities consist of same sub-activities but are executed in different order such as such as stacking objects and unstacking objects.

It should be mentioned that all experiments were implemented on a 2.5GHz Intel Core i7 PC with 4GB memory, running under Windows 8 Enterprise. The proposed system is coded using MATLAB 8.1.0.604 (R2013a). During the experiments, we used a cross-subject training/testing setup in which we take out each subject (i.e., leave-one-subject-out scheme) from the training set and repeat an experiment for each of them. This means the proposed system was trained on three of the four people from whom data was collected, and tested on the fourth. This is the same set-tings used in evaluating the state-of-the-art methods [16 - 21].

Figure 11 and Figure 12 show the confusion matrices of the proposed system using Cornell CAD-60 dataset and Cornell CAD-120 dataset respectively. Each row represents the instances in an actual class (groundtruth label) and each column denotes the recognition results. For example in the second row of Figure 11, 94% of the "brushing teeth" samples are classified correctly while 5% of the samples are misclassified as "rinsing mouth" activity and 1% are misclassified as "wearing contact lens" activity. As, it can be seen from Figure 11, the results prove the efficiency of the proposed method in recognizing differ-ent activities. The proposed system correctly classifies most activities. However, the perfor-

mance is rather degraded with very similar activities. For example, "talking on the phone" is confused with "drinking water" and "cooking (chopping)" is confused with "cooking (stirring)". Similarly, when we look at the confusion matrices in Figure 12, we can see higher values on the diagonal of the confusion matrix. They represent the activities that are correctly classified. The most difficult classes are "eating" and "scrubbing". "Eating" is sometimes confused with the "drinking", and "scrubbing" is likely to be confused with moving and placing. Also, "picking objects" is misclassied as "Taking food".

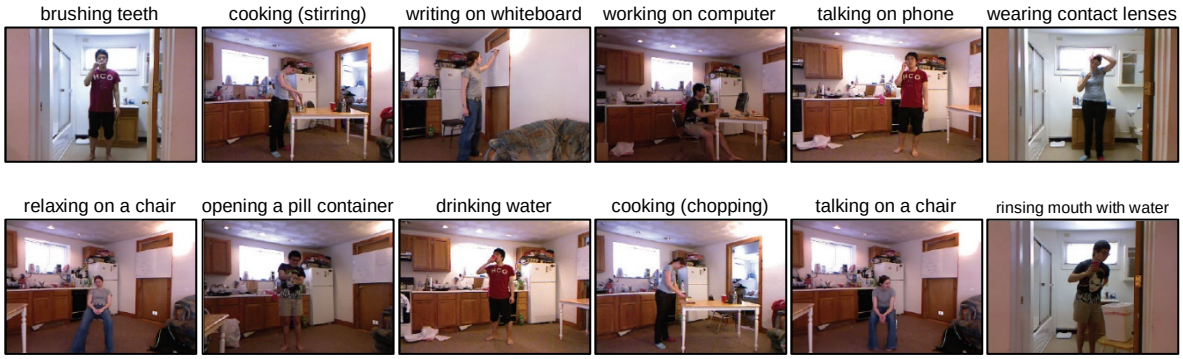


Fig. 9. Some example frames of Cornell CAD-60 dataset

Fig. 10. Some example frames of Cornell CAD-120 dataset sub-activities

TABLE 1
DESCRIPTION OF HIGH-LEVEL ACTIVITIES IN TERMS OF SUB-ACTIVITIES IN CORNELL CAD-120

high-level activities	sub-activities									
	reaching	moving	placing	opening	closing	eating	drinking	pouring	scrubbing	null
Making Cereal	√	√	√					√		√
Taking Medicine	√	√	√	√		√	√			√
Stacking Objects	√	√	√							√
Unstacking Objects	√	√	√							√
Microwaving Food	√	√	√	√	√					√
Picking Objects	√	√								√
Cleaning Objects	√	√		√	√				√	√
Taking Food			√	√	√					√

Arranging Objects	√	√	√									√
Having a Meal	√	√						√	√			√

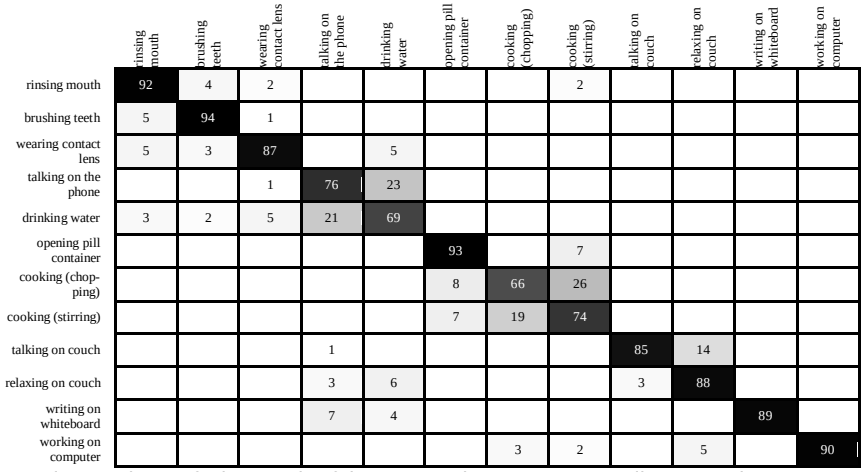
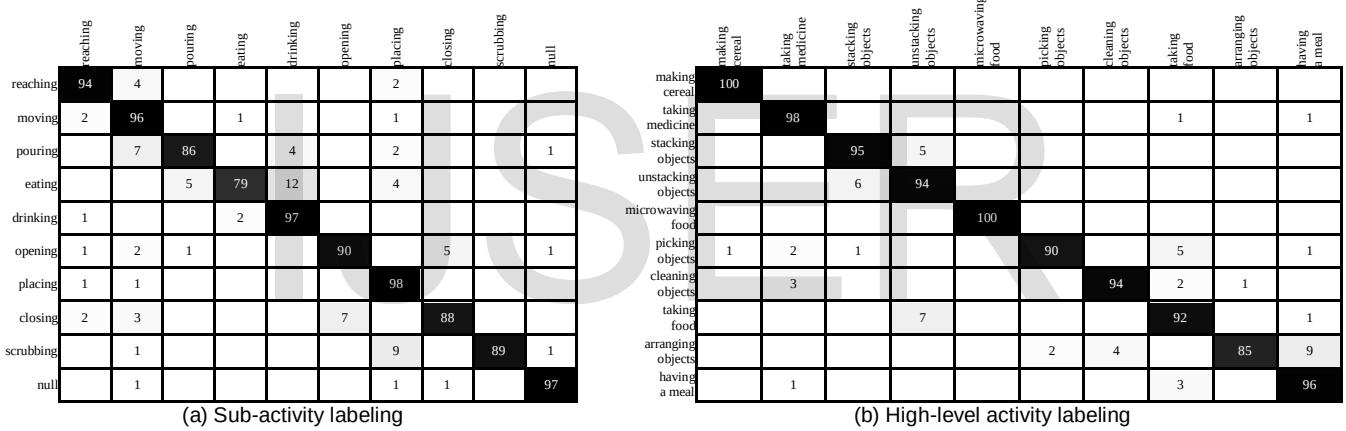


Fig. 11. The confusion matrix of the proposed system on Cornell CAD-60 dataset



(a) Sub-activity labeling (b) High-level activity labeling

Fig. 12. The confusion matrices of the proposed system on Cornell CAD-120 dataset

Moreover, we compare the performance of the proposed system with several recent methods [16-21]. Table 2 summarizes the comparative results of the proposed system and some of the state-of-the-art methods [16-18] on Cornell CAD-60. The proposed system achieves a precision and a recall rate equal to 83.4% and 81.2% respectively. Similarly, Table 3 shows the comparative results of the proposed system and another set of the state-of-the-art methods [19-21] on Cornell CAD-120. The proposed system achieves a recognition accuracy equal to 91.6% and 94.4% for sub-activity and high-level activity respectively. In [16-18], the performance results are reported in terms of precision, recall and $F_{0.5}$. However, in [19-21], the results are expressed in terms of recognition accuracy. Hence, we use the same measures to present the performance of our proposed system. It is obvious from the results shown in both tables that the proposed system performs better than several state-of-the-art methods. Also, note that the results achieved on Cornell CAD-120 are rather better than those obtained on Cornell CAD-60. This is because the number of video sequences in CAD 120 is twice the number in CAD-60 hence it gives the chance for training any recognition method more effectively.

TABLE 2
PRECISION AND RECALL SCORES (%) OF THE PROPOSED SYSTEM COMPARED TO THE STATE-OF-THE-ART METHODS ON CORNELL CAD-60 DATASET

Method	Cornell CAD-60 Dataset		
	Precision	Recall	$F_{0.5}$ (Average)
Sung et al. (2012) [16]	67.9%	55.5%	61.7%

Ni et al. (2013) [17]	75.9%	69.5%	72.7%
Gupta et al. (2014) [18]	78.1%	75.4%	76.8%
The proposed system	**83.4%**	**81. 2%**	**82.3%**

TABLE 3
RECOGNITION ACCURACIES OF THE PROPOSED SYSTEM COMPARED TO THE STATE-OF-THE-ART METHODS ON CORNELL CAD-120 DATASET

Method	Cornell CAD-120 Dataset	
	Sub-activity	High-level Activity
Koppula et al. (2013) [19]	86%	84.7%
Koppula & Saxena (2013) [20]	89.3%	93.5%
Hu et al. (2014) [21]	87.0%	NA
The proposed system	**91.6%**	**94.4%**

6 CONCLUSION

Even with great efforts made for the recent decades, the recognition of human activities is still an immature technology that attracted plenty of people. Recently, with the availability of inexpensive RGB-D sensors, the problem of human activities recognition has become relatively easier and more robust. However, most of these works only address detecting actions that stretches over short time periods not activities. In this paper, a skeleton-based human activity recognition system is proposed. The proposed system focuses on recognizing human activities not human actions. Human activities take place over different time scales and consist of a sequence of sub-activities (referred to as actions). The proposed system recognizes learned activities via trained Hidden Markov Models (HMMs). Experiments carried out on two benchmark datasets: Cornell CAD-60 and Cornell CAD-120. When compared to other skeletal-based solution our approach shows competitive performance.

REFERENCES

[1] Kavita V. Bhaltilak, Harleen Kaur, Cherry Khosla, "Human Motion Analysis with the Help of Video Surveillance: A Review," In the International Journal of Computer Science Engineering and Technology (IJCSET), Volume 4, Issue 9, pp. 245-249, September 2014.

[2] Chen Change Loy, "Activity Understanding and Unusual Event Detection in Surveillance Videos," PhD dissertation, Queen Mary University of London, 2010.

[3] Mao Ye, Qing Zhang, Liang Wang, Jiejie Zhu, Ruigang Yang, Juergen Gall, "A Survey on Human Motion Analysis from Depth Data," Lecture Notes in Computer Science, Springer Berlin Heidelberg, Volume 8200, pp 149-187, 2013.

[4] Lulu Chen, Hong Wei, James Ferryman, "A survey of human motion analysis using depth imagery," In Pattern Recognition Letters, Elsevier Science Inc., Volume 34, Issue 15, pp. 1995-2006, November 2013.

[5] Ronald Poppe, "A survey on vision-based human action recognition," In the International Journal of Image and Vision Computing, Volume 28, Number 6, pp.976-990, June 2010

[6] Maaike Johanna, "Recognizing activities with the Kinect," Master thesis, Radboud University Nijmegen, Nijmegen, Netherlands, July 2013.

[7] Ferda Ofli, Rizwan Chaudhry, Gregorij Kurillo, René Vidal, and Ruzena Bajcsy, "Sequence of the Most Informative Joints (SMIJ): A New Representation for Human Skeletal Action Recognition," In proceedings of the IEEE Computer Vision and Pattern Recognition Workshops (CVPRW), Providence, Rhode Island, USA, PP. 8-13, June 2012.

[8] Ahmed Taha, Hala H. Zayed, M. E. Khalifa and El-Sayed M. El-Horbaty, " Human Action Recognition based on MSVM and Depth Images," In The International Journal of Computer Science Issues (IJCSI), Volume 11, Issue 4, Number 2, pp. 42-51, July 2014.

[9] Xiaoxiao Dai, "Vision-based 3D Human Motion Analysis for Fall Detection and Bed-exiting," Master thesis, Faculty of the Daniel Felix Ritchie School of Engineering and Computer Science, University of Denver, USA, August 2013.

[10] Manjuatha M B, Pradeep kumar B.P., Santhosh.S.Y, "Survey on Skeleton Gesture Recognition Provided by Kinect," In the International Journal of Advanced Research in Electrical Electronics and Instrumentation Engineering (IJAREEIE), Volume 3, Issue 4, April 2014.

[11] Jungong Han, Ling Shao, Dong Xu, and Jamie Shotton, "Enhanced Computer Vision with Microsoft Kinect Sensor: A Review," In IEEE Transactions on Cybernetics, Volume 43, Number 5, pp. 1318 - 1334, October 2013.

[12] Vennila Megavannan, Bhuvnesh Agarwal, and R. Venkatesh Babu, "Human Action Recognition using Depth Maps," In proceedings of the International Conference on Signal Processing and Communications (SPCOM), Bangalore, India, pp. 1-5, July 2012.

[13] Samuele Gasparrini, Enea Cippitelli, Susanna Spinsante and Ennio Gambi, "A Depth-Based Fall Detection System Using a Kinect Sensor," In the International Journal of Sensors, Volume 14, Issue 2, pp. 2756-2775, February 2014.

[14] Ahmed Taha, Hala H. Zayed, M. E. Khalifa and El-Sayed M. El-Horbaty, "On Behavior Analysis in Video Surveillance," In Proceedings of the 6th International Conference on Information Technology (ICIT 2013), Al-Zaytoonah University of Jordan, Amman, Jordan, May 2013.

[15] Ahmed Taha, Hala H. Zayed, M. E. Khalifa and El-Sayed M. El-Horbaty, "Exploring Behavior Analysis in Video Surveillance Applications," In The International Journal of Computer Applications (IJCA), Foundation of Computer Science, New York, USA, Volume 93, Number 14, pp. 22-32. May 2014.

[16] Jaeyong Sung, Colin Ponce, Bart Selman, and Ashutosh Saxena, "Unstructured Human Activity Detection from RGBD Images," In Proceedings of the International Conference on Robotics and Automation (ICRA), Saint Paul, Minnesota, USA, pp. 842 - 849, May 2012.

[17] Bingbing Ni, Yong Pei, Pierre Moulin, and Shuicheng Yan, "Multilevel Depth and Image Fusion for Human Activity Detec-tion," In Cybernetics, IEEE Transactions, Volume 43, Issue 5, pp. 1383 - 1394, August 2013.

[18] Raj Gupta, Alex Yong-Sang Chia, and Deepu Rajan, "Human Activities Recognition using Depth Images," In Proceedings of the 21st ACM international conference on Multimedia (MM '13), Barcelona, Catalunya, Spain, pp. 283-292, October 2013.

[19] Hema S Koppula, Rudhir Gupta, and Ashutosh Saxena, "Learning Human Activities and Object Affordances from RGB-D Videos," In the International Journal of Robotics Re-search (IJRR), Volume 32, Issue 8, pp. 951-970, July 2013

[20] Hema S Koppula, and Ashutosh Saxena, "Learning Spatio-Temporal Structure from RGB-D Videos for Human Activity Detection and Anticipation," In

Proceedings of the Interna-tional Conference on Machine Learning (ICML), Atlanta, USA, June 2013.

[21] Ninghang Hu, Gwenn Englebienne, Zhongyu Lou, and Ben Krose, "Learning Latent Structure for Activity Recognition," In Proceedings of the International Conference on Robotics and Automation (ICRA), Hong Kong, China, pp. 1048 - 1053, June 2014.

[22] Xisheng He, Zhe Wang, Yingbin Zheng, and Xiangyang Xue, "A Simplified Multi-Class Support Vector Machine with Reduced Dual Optimization" In Pattern Recognition Letters Journal, Volume 33, Issue 1, pp. 71-82, January 2012.

[23] Xiaowei Yang, Qiaozhen Yu, Lifang He, and Tengjiao Guo, "The One-Against-All Partition Based Binary Tree Support Vector Machine Algorithms for Multi-Class Classification," In the Neurocomputing Journal, Volume 113, pp. 1-7, August 2013.

[24] Henry Joutsijoki, and Martti Juhola, "Kernel Selection in Multi-Class Support Vector Machines and its Consequence to the Number of Ties in Majority Voting Method," In Artificial Intelligence Review Journal, Volume 40, Issue 3, pp. 213-230, October 2013.

[25] Shian-Ru Ke, Hoang Le Uyen Thuc, Yong-Jin Lee,Jenq-Neng Hwang, Jang-Hee Yoo, Kyoung-Ho Choi, "A Review on Video-Based Human Activity Recognition," In the International Journal of Computers, Volume 2, Issue 2, pp.88-131, June 2013.

[26] Jaeyong Sung, Colin Ponce, Bart Selman, and Ashutosh Saxena, "Human Activity Detection from RGBD Images," In Proceedings of the AAAI Workshop on Pattern, Activity and Intent Recognition (PAIR), San Francisco, California, USA, August 2011.