Computer Vision and Image Understanding 140 (2015) 68–82

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Recognizing complex instrumental activities of daily living using scene information and fuzzy logic ☆

Tanvi Banerjee [a,*], James M. Keller [b], Mihail Popescu [c], Marjorie Skubic [b]

[a] Department of Computer Science and Engineering, Wright State University, 303 Russ Engineering Building, 3640 Colonel Glenn Highway, Dayton, OH 45435, United States
[b] Department of Electrical and Computer Engineering, University of Missouri, Columbia, MO, United States
[c] Department of Health Management and Informatics, University of Missouri, Columbia, MO, United States

ARTICLE INFO

Article history:
Received 24 July 2014
Accepted 14 April 2015
Available online 21 April 2015

Keywords:
Scene understanding
Activity analysis
Fuzzy logic
Activities of daily living
Eldercare
Depth images
Image features

ABSTRACT

We describe a novel technique to combine motion data with scene information to capture activity characteristics of older adults using a single Microsoft Kinect depth sensor. Specifically, we describe a method to learn activities of daily living (ADLs) and instrumental ADLs (IADLs) in order to study the behavior patterns of older adults to detect health changes. To learn the ADLs, we incorporate scene information to provide contextual information to build our activity model. The strength of our algorithm lies in its generalizability to model different ADLs while adding more information to the model as we instantiate ADLs from learned activity states. We validate our results in a controlled environment and compare it with another widely accepted classifier, the hidden Markov model (HMM) and its variations. We also test our system on depth data collected in a dynamic unstructured environment at TigerPlace, an independent living facility for older adults. An in-home activity monitoring system would benefit from our algorithm to alert healthcare providers of significant temporal changes in ADL behavior patterns of frail older adults for fall risk, cognitive impairment, and other health changes.

1. Introduction

Activities of daily living (ADLs) are a set of activities that are required for self-care such as walking, eating, dressing, and bathing. They are used to assess the functional capacity of older adults [11]. Instrumental ADLs (IADLs) are a subset of the functional tasks that older adults perform to support their independent lifestyles [9]. Examples of IADLs are housekeeping, cleaning, cooking. These activities, when measured over an extended period of time, can show deviations in health for older adults. Zisberg et al. [5] developed a new instrument called SOAR to evaluate routine patterns in the lives of older adults. Subjects from four retirement communities reported detailed information regarding ADLs like eating, meal preparation, watching television, bathing, etc. The study indicated that any deviation in the routine of frail older adults could correlate with a change in health and provides the motivation behind the work described in this paper. We describe the premise behind our study using the following case study revolving around the IADL *cleaning the table*. Suppose a healthy older adult living independently performs the IADL *cleaning the table* once every day at a certain time. However, due to some health related reason, she is unable to do so several days in a row. Once detected, this deviation from her normal routine could be a strong indicator of a health change which could help enable early interventions. The goal of this study is to build a model to learn these ADL or IADL patterns which can then be used for detection, and the changes in daily (or weekly or monthly) behavior patterns can then be used to detect early health changes.

The contributions of this paper are the following. We present a unique, vision-based method for recognizing components of ADLs and IADLs by combining their interaction with object surfaces with a set of linguistic fuzzy rules with heuristic parameters to model their activities. Specifically, in this paper we use the activities *walk*, *sit*, *clean object*, *clutter object*, *move near object*, *rearrange object*, and *move object* to describe our approach. We use the IADLs *make bed* and *eat* to describe the importance of combining scene information with moving object features to detect complex activities that are difficult to detect using only the foreground information or only the scene features. These activities further reinforce the importance of ontologies to provide context for each ADL or IADL that can provide the baseline for activity detection and help eliminate false alarms using contextual information. The results using our proposed algorithm are discussed and compared with another popular activity modeling algorithm, the

☆ This paper has been recommended for acceptance by Isabelle Bloch.

* Corresponding author.
E-mail addresses: tanvi.banerjee@wright.edu (T. Banerjee), kellerj@missouri.edu (J.M. Keller), popescum@missouri.edu (M. Popescu), skubicm@missouri.edu (M. Skubic).

http://dx.doi.org/10.1016/j.cviu.2015.04.005
1077-3142/

hidden Markov model (HMM) and its variation, the details of which are provided in Section 8. We further test our method on data collected in an apartment at TigerPlace, an independent living facility for older adults. The data comprise depth information from an older resident (age 88, without any ambulatory needs such as a walker) as he goes through his daily routine in the apartment. We conclude with the discussion of the future steps for the ADL activity modeling framework. The next section reviews some of the related work in this field using vision and non-vision based sensors.

2. Background

Studies described in [4,5] indicate the importance of longitudinal analysis of the daily routine of older adults to study anomalies or deviations in their regular patterns in an automated, non-intrusive manner. In order to detect these deviations, the activities need to be recognized and ordered in a methodical way for day-to-day behavior comparison. One approach is based on ontological activity modeling. This is described in more detail in the next section.

In related activity modeling work using sensor information and probabilistic approaches, the researchers in [22,23] utilized motion sensor data to learn context-aware rules using a Bayesian network (BN). The ADLs tested were personal hygiene, bathing, toilet transition, housekeeping, eating, leaving home, sleeping, and taking medication on two residents; the resulting activity label accuracy was approximately 70%. In [34], the researchers also used BNs to learn a specific ADL, brushing teeth, using a combination of camera and motion sensors.

In work using vison-based sensors, Pirsiavash and Ramanan [35] used a wearable camera to detect objects of interest to identify 18 different ADLs using a combination of bag of words approach with an object detection model. However, this technique relied strongly on the ability of the algorithm to recognize different objects such as water faucet, oven, etc. Also, it is not realistic to expect older adults to wear cameras while they go through their normal routine. In work using fuzzy logic, Brulin et al. [32] detected the simple activity states of lying, squatting, sitting, and standing using a set of fuzzy rules. They also had a state called "undetermined" to identify unknown activity states. Simple bounding box parameters from the silhouettes obtained from a single camera were used as input to the single layered, eight ruled, fuzzy rule based system. Accuracy results range between 64% and 72% depending on the dataset. Their work, similar to the study in [10] focused on detecting falls in an in-home environment and not on the ADLs performed by older adults in their normal routines. In other works related to depth data, there are several studies to detect different ADLs [31,33,36–38]. However, all of these studies utilize both color and depth information to detect the ADLs. To our knowledge, there has been no work on ADL detection using only depth information from the Kinect sensors. We have chosen to restrict the data to depth images only due to privacy concerns. Prior research has shown that seniors are willing to accept the use of silhouette imagery even though they consider continuous RGB video monitoring to be a privacy invasion [39]. The techniques proposed here rely on segmented 3D silhouettes for ADL recognition.

In a review paper, Lavee et al. [24] described the different methods of activity event detection with vision-based sensors using pixel-based, object-based, and logic-based approaches. For pixel-based approaches, they described techniques using color, texture, as well as gradient information. For object-based approaches, they described features such as bounding box and speed of moving objects. For logic-based approaches, they described techniques that use rule-based activity models. Our approach incorporates features from the moving person as well as from the scene using depth data to build a robust activity model framework that can handle uncertainties of activities being performed in different ways. We illustrate this variance with the following scenario. Consider two residents, A and B. The normal routine for resident A having lunch is as follows: he goes to the refrigerator, gets some deli meat and cheese, makes a sandwich, and then sits at a dining table to eat his sandwich. The normal routine for resident B having lunch is as follows: she opens a can of soup from the cabinet, heats it on the stove and then eats in the living room. As can be seen, there are variations in the same activity *eating lunch* between different individuals. A robust activity model needs to be able to handle these variations within the same activity and still be able to identify both instances.

Another common approach to activity modeling is the use of HMMs as an event modeling formalism. An HMM is a doubly stochastic process, i.e. there is an underlying stochastic process that is not observable (hidden) but can only be observed through another set of stochastic processes that produce the sequence of observed symbols [20]. Several studies, including [18,25,26], use HMMs to detect activities such as meal preparation, eating snacks, and washing dishes. We will use this method for comparison with our activity framework.

3. Ontological framework

The idea for representing ADLs using an ontology is not new. In [6], Chen et al. proposed an ontological method to recognize ADLs such as housework, managing money, taking medicine, and using the phone. Theoretical foundations were set up to fuse information from different sensors (contact sensors, motion sensors, tilt sensors and pressure sensors), and then build an ontology of ADLs. Data from all the sensors were aggregated to describe the ADL occurring at a certain time point. Experiments were conducted under laboratory settings and tested on a subset of the ADL activities including brushing teeth, bathing, and watching television. An accuracy of 94% was achieved on a small subset of three subjects. In another study, Latfi et al. [7] described an ontological approach to describe the medical history of older adults in an assisted living facility using a system called Telehealth Smart Home system (TSH). In this framework, they created an ontology which comprised the person and his/her medical history. The person component contained the profile of the person, interactions with the staff, and other interactions on a social level. The medical history comprised the individual's deficiencies (physical, sensory), diseases, and risk factors. In ontologies related to eldercare technologies, Rodríguez et al. [8] proposed a framework called CARe to describe ADLs in a nursing home scenario. Similarly, the researchers in [42] proposed a framework called ELDeR to support independent lifestyles of older adults. In other ADL work, the study in [40] described ADLs using specific examples such as nocturnal activities, and the study in [41] gave an overview of ADL ontologies in the context of smart home applications. However, none of these studies were implemented or tested in a real smart home environment, and only theoretical foundations were mentioned.

We now describe our ontological framework for learning ADLs in general. One unique aspect of our approach is that it looks at the overall big picture of the ADL framework *while still being able to handle incomplete information*. Fig. 1 shows the ontological structure for the activities. There are five categories: activities, location, objects, sensors and time of day. The activities component can be atomic or complex. The atomic activities such as upright, walking, sitting, bending, and rising are the building blocks of complex activities. The complex activities comprise the ADLs and IADLs. The location parameter describes the locations inside the apartment which can provide context to the activity taking place. For example, making the bed is most likely to take place in the bedroom. The objects component refers to the objects with which seniors interact to perform the ADLs and IADLs. The sensors component describes the sensors in our smart home system used to detect the behavior patterns of older adults in their home setting. The final component, time of day, refers to the time when the activities take place. This is useful to learn the patterns of the behavior trends of older adults on a daily basis. For our study,

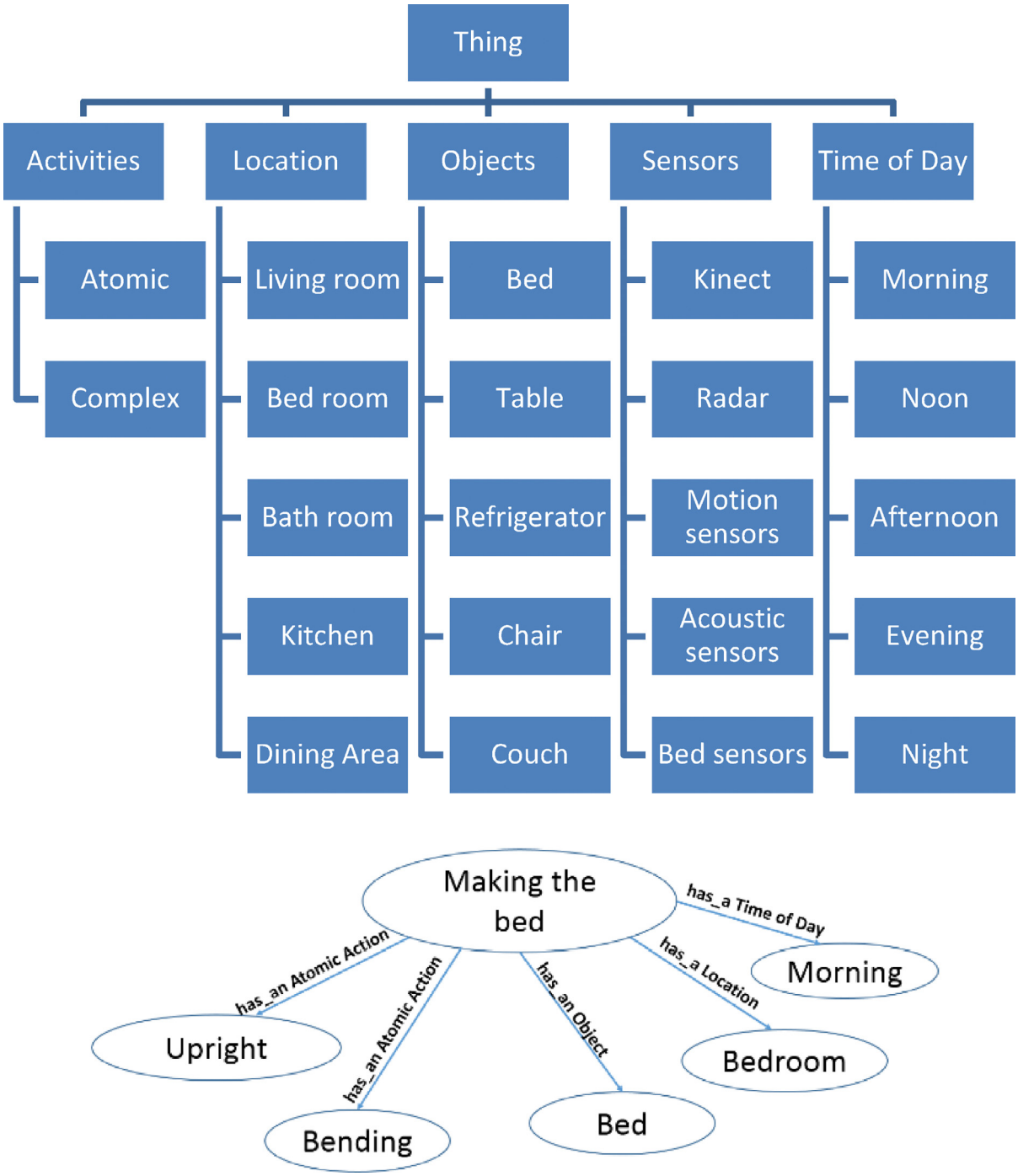

Fig. 1. Top: general ontological framework for activities. Bottom: example of the IADL making the bed.

we have used the depth camera as the sensor, and used manually labeled object surfaces as well as room information to identify specific IADLs. The components from the ontology described in Fig. 1 provide the critical contextual information to describe ADLs and IADLs. For an automated ADL and IADL recognition system, there will be a need to incorporate an object surface recognizer that can identify specific instances of the ADLs. However, we propose a method to handle incomplete information that can be very useful in a dynamic environment where all the object surfaces may not be identified and models for all possible ADLs and IADLs are not created. This technique is described in the next few sections and elaborated in Section 10.

An example of an IADL "making the bed" is shown in Fig. 1 which describes some of its attributes. For example, it has an atomic action bending, has an object bed, has a location bedroom, etc. Our framework differs from other ontological models in its ability to handle uncertain events. Instantiation of different activities is a daunting task, especially when the same activity can be performed differently by different individuals. By instantiation, we mean that we annotate or explicitly identify a specific ADL or IADL such as "making the bed". In this case, we utilize the underlying assumptions regarding the activity; such as it occurs in the bedroom, the object surface is the bed, and the conditions shown in Fig. 1. Our model can handle these uncertainties while still being able to provide information

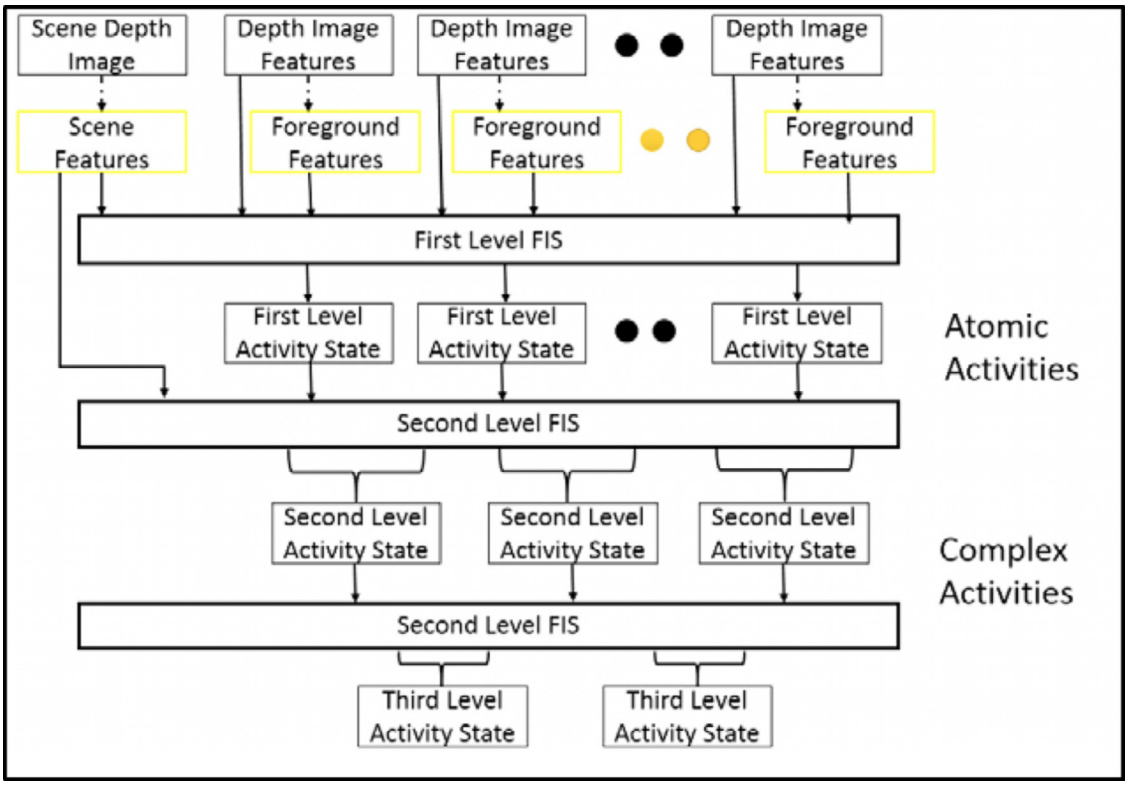

Fig. 2. Block diagram of the activity modeling system from depth video sequences using a fuzzy inference system.

which can be used to detect behavior trends. The other difference is that we have tested part of the ontology in an independent living senior housing and demonstrate the efficacy of our research. Fig. 2 gives the block diagram of our system. Only depth information is used to get information of the moving foreground as well as to obtain scene information. The features extracted are then input to the first level FIS which provides the first level activity state information that corresponds with the atomic activity states described in our ontological framework. This information is then further coupled with automatically-extracted scene information and input to the second level FIS, which then generates summaries of complex activity states. Finally, these summaries are further input to the third level FIS to generate the final activity summaries. Level 1 corresponds to the atomic level activity states, level 2 produces summaries incorporating interaction with the objects present in the scene, and level 3 produces IADL and ADL summaries using the second level summaries for more complex activity recognition. Consider the following scenario. Suppose a person is detected sitting at the table. If we add another layer to the activity states, we can see what happened prior to the event. If there is detected movement near the kitchen or the refrigerator, it is likely that he is having a snack. The combination of the two activities (initial movements in the kitchen with specific object surfaces, followed by sitting at the table) improves our confidence that the IADL activity is eating and reduces false alarms by adding more context to the event.

Our approach to monitoring human activity is based on fuzzy set theory. One of the advantages of using fuzzy logic for activity modeling is the small sized training data needed to extract the FIS parameters. Since a fuzzy rule-based system comprises a set of rules that are determined heuristically, there is no need for a large training data set. That said, we do require some training data to determine the membership function parameters, which is discussed in Section 7. Another important advantage of the FIS is the ability to describe the model linguistically. This is especially useful in our interdisciplinary environment where our clinical partners (nurses, physicians, physical therapists, and social workers) prefer to interpret the output of the model in order to diagnose the residents for early intervention using their activity information. We now describe our sensing modality and data capture process in the next section.

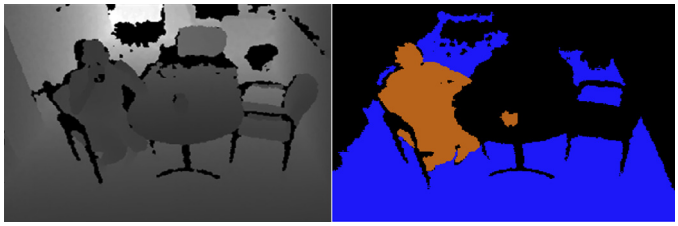

Fig. 3. Example of a depth image (original size 244 × 320 pixels) and its corresponding foreground image. The blue area in the foreground image is the ground region and the large orange object is the detected moving object (a person). The smaller orange object is an object (a bowl) on the table placed by the person which is detected as foreground since it was not a part of the learned background model. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

4. Depth video segmentation

Foreground is extracted on the raw depth images from a single Microsoft Kinect sensor using a standard background subtraction algorithm. The background is learned using the mixture of Gaussian approach. Any depth value outside this range is recognized as a foreground pixel [2]. We utilize a dynamic background update algorithm to account for the constant changes in the environment in real world settings. The ground plane in the Kinect's field of view is extracted as in [2]. Ground points are selected manually during the sensor setup

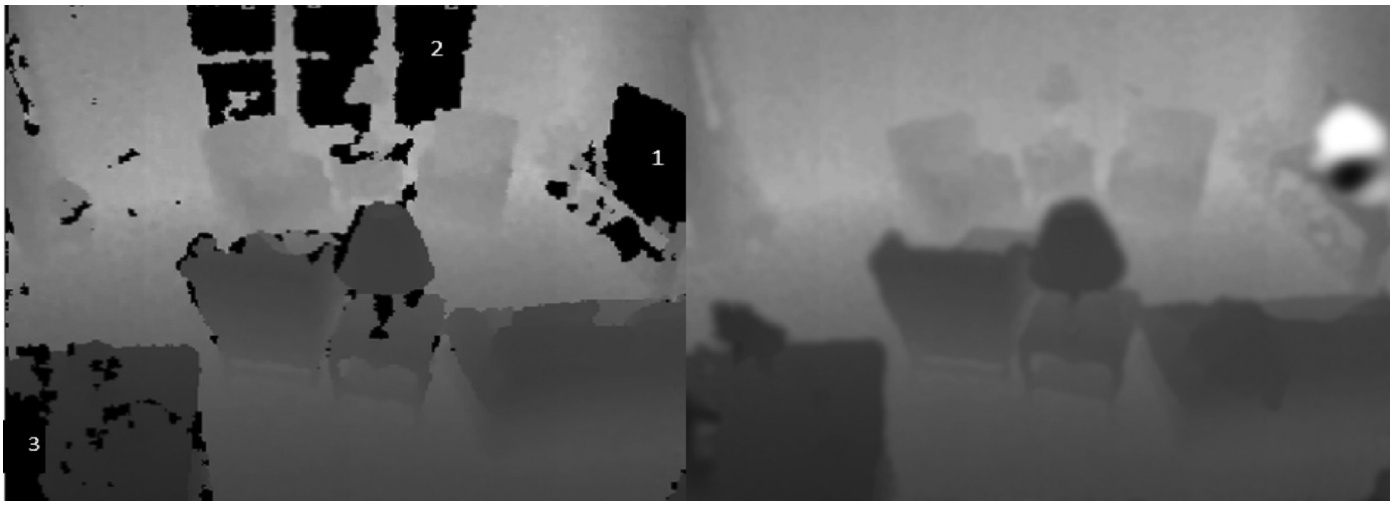

Fig. 4. Example of a depth image and its corresponding inpainted image.

and the ground plane is estimated in an iterative manner using the RANSAC plane fitting method approach described in [15]. An example of a depth image and its corresponding foreground image is shown in Fig. 3. A sample video is provided in the url provided.[1] The advantage of depth sensors is that, unlike vision sensors, their performance remains unaffected under low light conditions or even in the dark. The sensor is positioned in the living environment and does not require any wearable components. Using only the depth data from the Kinect sensor protects the person's privacy since we do not use or store the color images and only shape information is extracted from the depth data, i.e., effectively a three-dimensional silhouette.

5. Scene understanding

This section describes our method to obtain surface information from the scene. Prior to feature extraction, we employ a region filling operation to remove noisy depth pixels from the image described in Section 5.1.

5.1. Image inpainting

Most of our previous activity detection algorithms using vision sensors are based on analysis of foreground objects [1,2]. This could be as simplistic as tracking the centroid positions of a moving person to measure the walking speed [2] or using shape descriptors to provide further insight into the activity state [12]. The work proposed here uses the cues provided by the scene itself to recognize the activity of a person. However, in order to identify these cues, the image has to be filtered. The first step in this process uses a region filling algorithm which removes random noise generated by the depth sensor. For the video inpainting algorithm, we use the technique described in Telea [3]. For our application, the regions to be filled are the black regions in the image depth image, i.e. all the areas which do not return any depth values to the sensor (where the infrared emissions do not reflect back to the sensor). Fig. 4 shows a raw depth image of a room obtained from the Kinect sensor on the left, and the cleaner image after inpainting on the right. We can see that some of the artifacts on the table (left) and the windows (top) disappear after the inpainting technique.

5.2. HONV features and horizontal surface extraction

For the next step, we compute the Histogram of Oriented Normal Vectors (HONV) for all the pixels in the inpainted image.

Table 1
Comparison of dense SIFT (dSIFT) with the HONV features.

Scenario	Number of horizontal surfaces	dSIFT detected	dSIFT FP	HONV detected	HONV FP
1	5	4	0	5	0
2	4	4	0	4	0
3	5	4	0	5	0
4	4	4	0	4	0
5	5	5	1	5	0

Unlike other color image based features such as dense SIFT [1,14], the HONV features [13] were specifically created for depth images to provide a description of the local structural features in the depth image. For every patch in an image, the HONV feature computes the normal vector to that region using depth values. Since depth images return distance values of different objects present in the scene from the sensor, the normal vector obtained at each of these pixel locations from the image provides gradient information about surfaces that can highlight the similarities or disparities of surfaces in the field of view. For each pixel, the orientation is quantized into 8 bins for window size of 4×4. Then, the top three principal components are retained. Fig. 5 shows the identified horizontal surfaces (pink) extracted using the top three principal components of the (b) the dense SIFT feature and (c) the HONV vector on the inpainted image (Fig. 5a). The window size as well as the quantization bin values for the dense SIFT computation were kept the same as the HONV for a fair comparison. The regions in the image whose normal vectors are within the range of the minimum and maximum values of the normal vectors from the extracted ground region (method described in Section 4) are highlighted in pink. These are the identified horizontal surfaces labeled to show the number of detected surfaces. The ground region is excluded from the surfaces since we are detecting horizontal surfaces from objects above the ground surface.

As shown, the surface labeled 2 in the HONV image (Fig. 5c) is missing in the dense SIFT image (Fig. 5b). Comparison using five different scenes (images posted in the link[2]) is presented in Table 1.

We can see that all the surfaces were detected accurately using the HONV features. We found the HONV features to work better for surfaces that are further away from the depth camera as compared to the dense SIFT features. Also, the processing time is much faster than

[1] http://www.eldertech.missouri.edu/adl/

[2] http://www.eldertech.missouri.edu/adl/

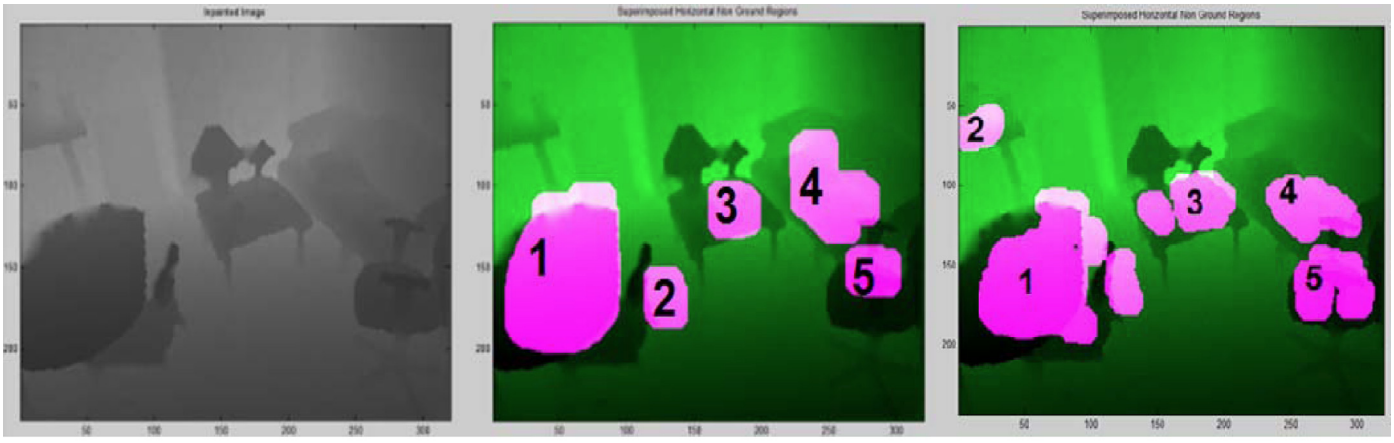

Fig. 5. Horizontal surfaces (labeled 1–5) obtained from the inpainted image (left) using the SIFT features (b) and HONV features (c).

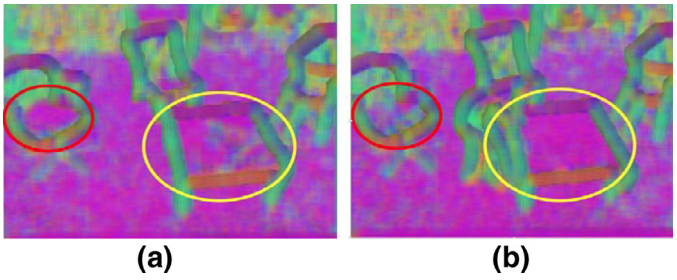

Fig. 6. Detecting clutter on the surfaces of the chair circled in red (left) and the table (right) circled in yellow (a) before activity and (b) after activity using the HONV images (top three PCA components described in Section 5.2). The chair is cluttered more after the event and the table is cluttered less which can be used for activity inference as will be described in the next section. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

the dense SIFT method (approximately half the time). For a 244 × 320 depth image, it takes around 0.8 s to compute the HONV and around 1.56 s for the dense SIFT using a six-core i7 machine at 3.4 GHz with Windows 7 (Fig. 6 and Tables 2 and 3).

6. Features for the fuzzy rule based system

In this section, we describe the features we input to our hierarchical system of fuzzy inference for activity reasoning. Once the areas of interest are extracted, features are computed for activity recognition. For a given sequence, these features are calculated only if a moving object, hereafter labeled as the Assumed Person (AP) to distinguish it from scene objects, is detected. This helps to speed up processing so that only sequences with noticeable movement are further considered for activity analysis. For these experiments, the rules and membership values are developed heuristically using a small training sequence. Rules are added in an iterative manner and those which do not improve the performance significantly are removed.

6.1. Bounding box features (image plane features)

These are the parameters of the minimum dimension rectangle that can be fit to the APs. The three features are the width (BBX), and area (BBA) of the rectangle. The temporal differences of BBA and BBX are also considered, termed DBBA and DBBX, respectively.

Table 2
Rule for near object activity state.

	NB	DO	DS	H	DH	Near object
1	L	L	M	H	Z	H

Table 3
Rule for field of view activity state.

	NB	Field of view
2	H	H

6.2. Centroid position (volumetric feature)

This is the (x, y, z) location of the moving AP. For input to the FIS system, the difference in these values in consecutive frames is computed (temporal differencing). These are called DXY (difference in XY location) and DH (difference in height), respectively. The height of the foreground object (H) is also an input to the first level FIS.

6.3. Distance from the object (volumetric feature)

This is used to detect the proximity to the object surface. For input to the first level FIS, the smallest distance between the foreground object and the object surface (DO) is considered.

6.4. Distance from the sensor (volumetric feature)

This is used to detect the proximity to the Kinect sensor. We consider this input since the depth values are inaccurate when the detected foreground is too close or too far to the sensor. Studies such as [45], as well as our own experiments indicate that depth values obtained from the Kinect sensor are sensitive to the distance of the objects in the field of view to the sensor. If an object is too close or too far to the Kinect, the depth values from the sensor get distorted so that needs to be taken into account in a robust activity recognition framework. For input to the first level FIS, the distance between the centroid of the AP and the sensor (DS) is considered.

6.5. Number of points on the boundary (image plane feature)

This parameter is important to determine the confidence of the person exiting/entering the field of view. The number of points on the boundary (NB) is measured by the number of points of the AP present on the field of view boundary. This is defined as the number

of pixels of the AP in the boundary of the depth image plane. More description can be found in our earlier work described in [16].

6.6. Orientation (image plane feature)

Orientation of the moving AP is computed as the angle (in degrees ranging from −90 to 90°) between the x-axis and the major axis of the ellipse that has the same second-moments as the region with the centroid of the AP as the reference point. For example, the orientation of a horizontally oriented AP is close to 0°; the orientation of a standing person would be close to 90°. This input feature is labeled as O to the FIS.

6.7. Change in ground plane (image plane feature)

This feature detects any changes in the background environment before and after an activity event. Specifically, if an object such as a chair is moved, there is a change in the visible ground plane from the sensor's field of view. Eqs. (1) and (2) describe the images used to compute this change. Eq. (1) shows the before-background image (before the event) as the union of the frames in the first 2 s of the event using the max operator. This ensures that for a given pixel, only the farthest value is preserved so that if there are moving objects present in the scene, they are removed. Similarly, Eq. (2) shows the after-background image using frames from the last 2 s of the detected event. The event is identified by any one of three conditions as described in Section 7.1.

$$\text{Before } BG = \cup_{\text{frames in first 2 s}} D(x, y) \tag{1}$$

$$\text{After } BG = \cup_{\text{frames in last 2 s}} D(x, y) \tag{2}$$

The change in ground plane information (CG) then is the absolute value of the difference in the number of detected ground plane pixels between these images.

6.8. Degree of clutter and degree of overlap (image plane features)

The next feature, called degree of clutter of an object surface (DC), is added to look at what happened to the surfaces before and after a specific activity involving object interaction. Consider the activity *cleaning table*. If the subject is cleaning the table, he/she will remove or rearrange the table surface to clean it. We can use the change in the table surface information to gain more insight into the activity. This is computed as the number of pixels on the horizontal surfaces (Section 5.2). This parameter refers to how "flat" the horizontal surface is, i.e. the more horizontal the surface, the larger number of horizontal pixels present on the surface. The DC is then the difference in the number of horizontal pixels detected on the object surface before and after an activity event. For this, we again use the images BeforeBG and AfterBG (Eqs. (1) and (2)). For normalization, the number of pixels on a horizontal surface is divided by the size (area of bounding box of the surface) of the object. The degree of overlap (DOO) is the number of horizontal pixels in common between the BeforeBG and AfterBG images. This determines the change in position of the objects on the surface of an object. This is also normalized by the object size in the AfterBG image (using the bounding box of the surface).

7. The fuzzy inference system

The features described above are input to the three-layered FIS for automated reasoning. Our approach to monitoring human activity is based on fuzzy set theory [28] which is an extension of classical set theory. One of the more well-known branches of fuzzy set theory is fuzzy logic [29]. Fuzzy logic is a powerful automated reasoning framework which comprises an inference system that operates on a set of rules structured in an IF-THEN format (example: "IF X is A, THEN Y is B"). The IF part of the rule ("IF X is A") is called the antecedent, while the THEN part of the rule ("THEN Y is B") is called the consequent. Here, X and Y are input and output variables, respectively while A and B are linguistic values that can be interpreted by humans (e.g. small, medium, large). These linguistic values can be defined using membership functions that map any input domain to the real-valued interval [0, 1]. These functions can be expressed in different forms such as triangular, trapezoidal, Gaussian that represent the degree to which the input fits the specific function. We will look at an example in Section 7.1. In this work, we use the standard Mamdani fuzzy inference system [29,30]. The system has three levels of fuzzy rules; the first level involves acquiring the confidences in atomic activity states. The second level of fuzzy logic performs activity recognition from features of the first level for further complex activity analysis, and the final third level is used for further activity inference to obtain IADL and ADL recognition. The linguistic summaries generated as output from the FIS have the advantage of being understandable by a human while achieving our goal of automated activity reasoning.

Table 4
Rules for bend near object activity state.

	DO	DXY	DS	BBX	BBA	H	DH	Bend near object
3	L	L	M				N	H
4	L		M	NVH	NVH	M	N	H
5	L		M	NVH	NVH	M	N	H
6	L	VL	M				N	H
7	VL	L	M				N	H
8	VL		M	NVH	NVH	M	N	H
9	VL		M	NVH	NVH	M	N	H
10	VL	VL	M				N	H

7.1. First layer FIS

This section describes the rules which determine the confidence for atomic activity states of *near object*, *at boundary of field of view* (of the sensor), *downward motion near object* (bend near object), *upward motion near object* (rise near object), *move object*, *on object*, *on object horizontal* (on object hor), *walk* and *previous state*. By *previous state*, we mean that the same activity state as the previous frame is continuing in the current frame. The rule used to detect this is provided in Table 8. When we describe events involving interaction with an object, we specifically mean interaction with the horizontal object surface. For the rest of the paper, we will use this terminology. For our current experiments, 37 rules are used. In the interest of computational efficiency, we kept the rule base as small as possible. These rules are fired when at least one of the following three conditions are satisfied:

- When someone enters or exits the field of view.
- When the time elapsed is greater than 2 min and someone has been in the field of view during the time period.
- When there is change in height below 10 in. from the 3D depth information of the moving objects.

We categorize the rules into nine tables, one for each of the nine atomic activity states. For the first group, the five parameters used are NB, DB, DS, H and DH (described in Section 6) with trapezoidal or triangular membership functions. The membership functions for the input features are shown in Fig. 7. The membership functions for all the output variables are shown in Fig. 8. All the antecedents are joined using the AND operator as a connective for the rule generation.

Table 4 provides the rules for low-level activity state *bend near object*. Note that the fuzzy membership not very high (NVH) is defined as the complement of VH i.e. NVH = 1 − VH. Hence, the fuzzy membership function for that is the complement of the membership function for VH (Table 5).

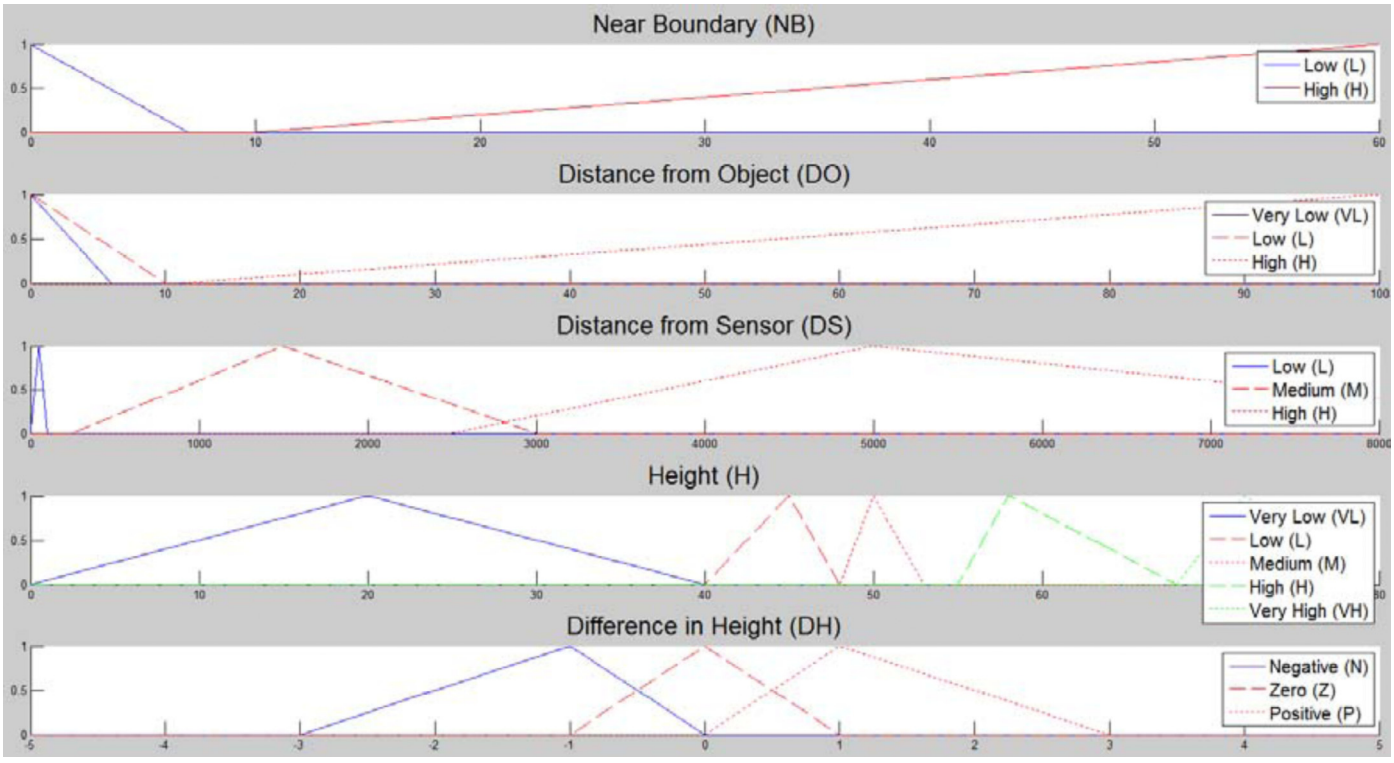

Fig. 7. Membership function plot for input features NB, DO, DS, H, and DH. Here, the *Y* axis is the membership value ranging from 0 to 1.

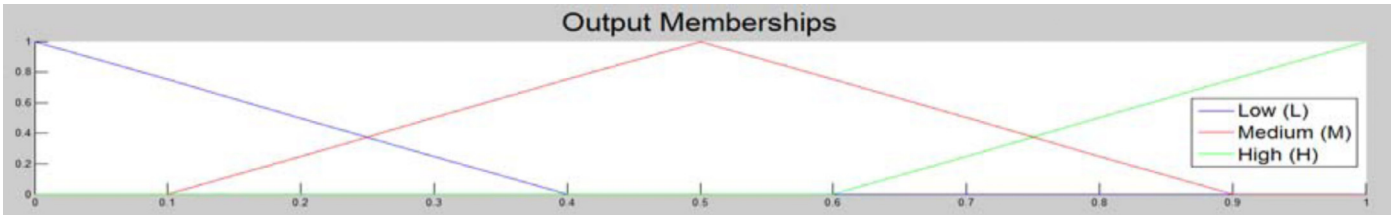

Fig. 8. Membership function plot for output variables. Here, the *Y* axis is the membership value ranging from 0 to 1.

Table 5
Rules for rise near object activity state.

	DO	DXY	DS	BBX	BBA	H	DH	Rise near object
11	L	L	M				P	H
12	L		M	NVH	NVH	M	P	H
13	L		M	NVH	NVH	M	P	H
14	L	VL	M				P	H
15	VL	L	M				P	H
16	VL		M	NVH	NVH	M	P	H
17	VL		M	NVH	NVH	M	P	H
18	VL	VL	M				P	H

Table 6
Set of fuzzy rules for level 1.

	DXY	DS	H	O	DH	On object
19	L	M	VL	P	N	H
20	L	M	VL	P	N	H
21	L	M	VL	P	Z	H
22	L	M	VL	N	N	H
23	L	M	VL	N	N	H
24	L	M	VL	N	Z	H

Table 7
Set of fuzzy rules for level 1.

	DO	DXY	DS	H	O	DH	On object Hor
25	L	L	M	VL	NU	N	H
26	L	L	M	VL	NU	N	H
27	L	L	M	VL	NU	Z	H

Fig. 9 shows the membership functions for DXY, BBX, BBA, DBBA, DBBX, O, and CG.

Table 7 gives the rules for *on the object horizontal* state. If the orientation of the foreground object is not horizontal, then it will generate a stronger confidence for the rules from Table 6. Otherwise it will have a stronger confidence for the rules from Table 7.

Table 8 gives the rules for evaluating the confidence for *move object*. *Move object* is part of cleaning the house or the housekeeping IADL and is also an identifier of scene change. Here, moving an object is identified by a sudden increase in the area and the width of the bounding box features while the person is near the object. The linguistic interpretation of Rule 29 is: ***IF*** the Distance from Object (DO) is ***High*** and the Difference in Bounding Box width (DBBX) is ***Low*** and the Distance from Sensor (DS) is ***Medium*** and the Change in Ground (CG) is ***High***, ***THEN*** the membership for **Move Object** is *Very* ***High***.

Previous state indicates that there is no significant change in activity state as compared to the previous frame. The rule used to determine this is provided in Table 9.

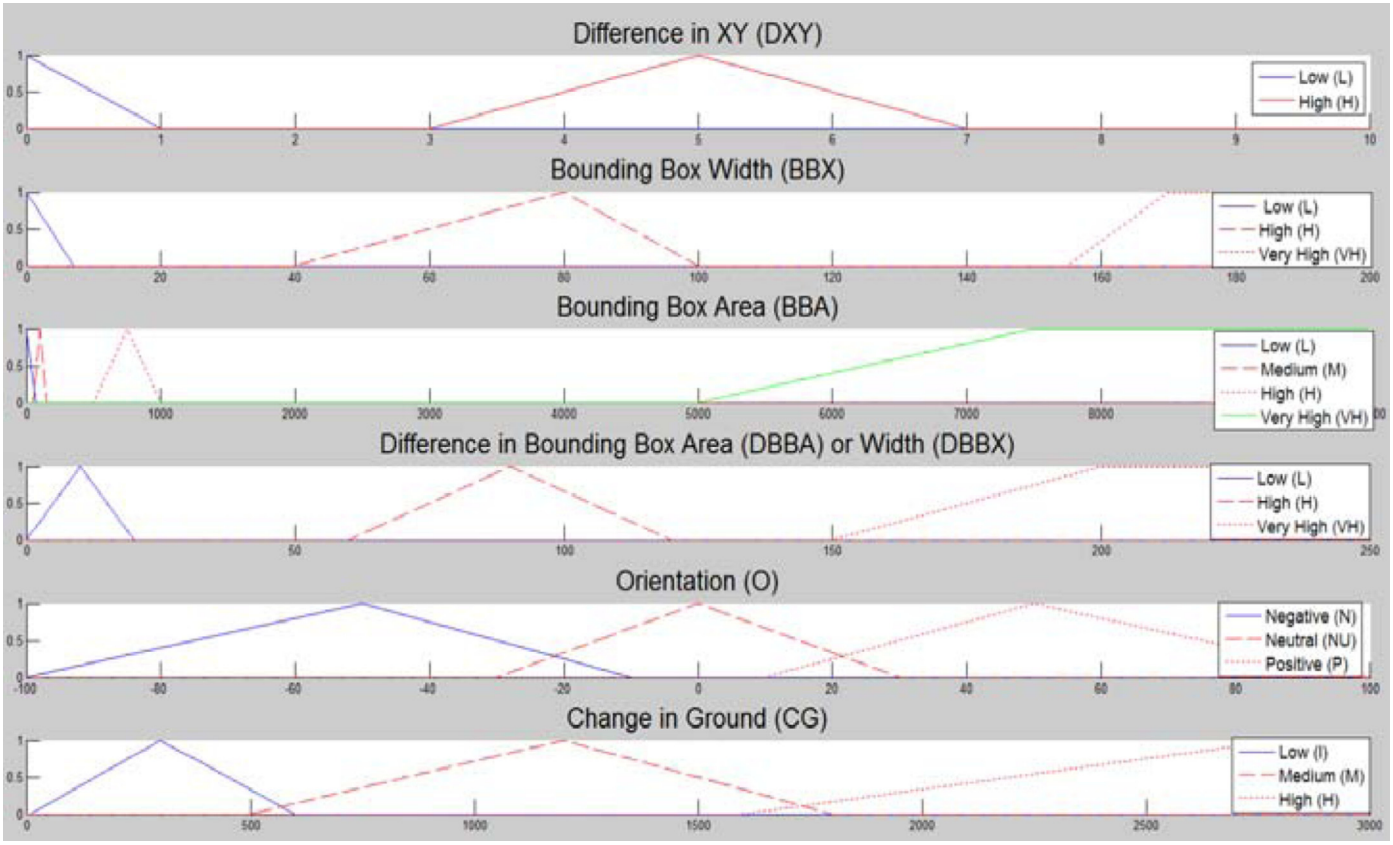

Fig. 9. Membership function plot for the next set of input features.

Table 8
Set of fuzzy rules for activity state move object.

	DO	DBBX	DS	CG	Move object
28	L	H	M	M	H
29	L	H	M	H	VH
30	L	M	M	M	H
31	L	M	M	H	VH
32	L	H	M	M	H
33	L	H	M	H	VH
34	L	M	M	M	H
35	L	M	M	H	VH

Table 9
Set of fuzzy rules for activity state previous state.

	DH	DXY	Previous state
36	VL	VL	H

Table 10
Set of fuzzy rules for activity state walk.

	H	DXY	Walk
37	H	H	H

Table 10 gives the simple rule to detect walking. Here, the idea is that if the height of the moving object is high and the change in the *x–y* coordinates from the previous frame is high, then there is a strong confidence of the activity state being a *walk*.

Using the above set of rules, we can obtain the confidence of the atomic activity states for the individual frames. Individual atomic states were detected only when the confidence values exceeded 0.5. This also takes care of the condition when there are no rules fired since in that case, the default value for the states is 0.5. Summaries are then generated for each of these atomic activity states after temporal filtering using a window size of three frames (about 0.5 s). The information is stored by retaining the beginning and end time of that state, as well as the confidence value for that summary (max of the confidence values in that time interval). These atomic states are then used for the second level activity segmentation. While this is not a complete set of rules by any means, the rule set is able to identify the states of the small training dataset. If we were to evaluate a complete set of all the possible combinations of the fuzzy membership functions, there would be at least 14^5 rules as compared to the 37 rules currently implemented. This would not only increase the computation time exponentially but also make the system difficult to adapt if we tried to learn all the parameters using artificial intelligence techniques.

7.2. Second layer FIS

The next level of fuzzy logic performs activity recognition using features computed in the first level. For this stage, we primarily focus on four general activity states involving horizontal surfaces: clean, clutter, rearrange and vertical movement near the horizontal surface. The features we use to determine the activity have been described previously in Section 6.8 termed degree of clutter (DC) and degree of overlap (DOO). When there is an increase in the degree of clutter, the high-level activity state may be *clutter object*. Some more features from first level activity inference need to be satisfied to increase our confidence for *clutter object* activity. Similarly, when there is a decrease in the degree of clutter, the high-level activity state can be *clean object* and if there is no significant change in the degree of clutter, the high-level activity state can be *rearrange object*. This means

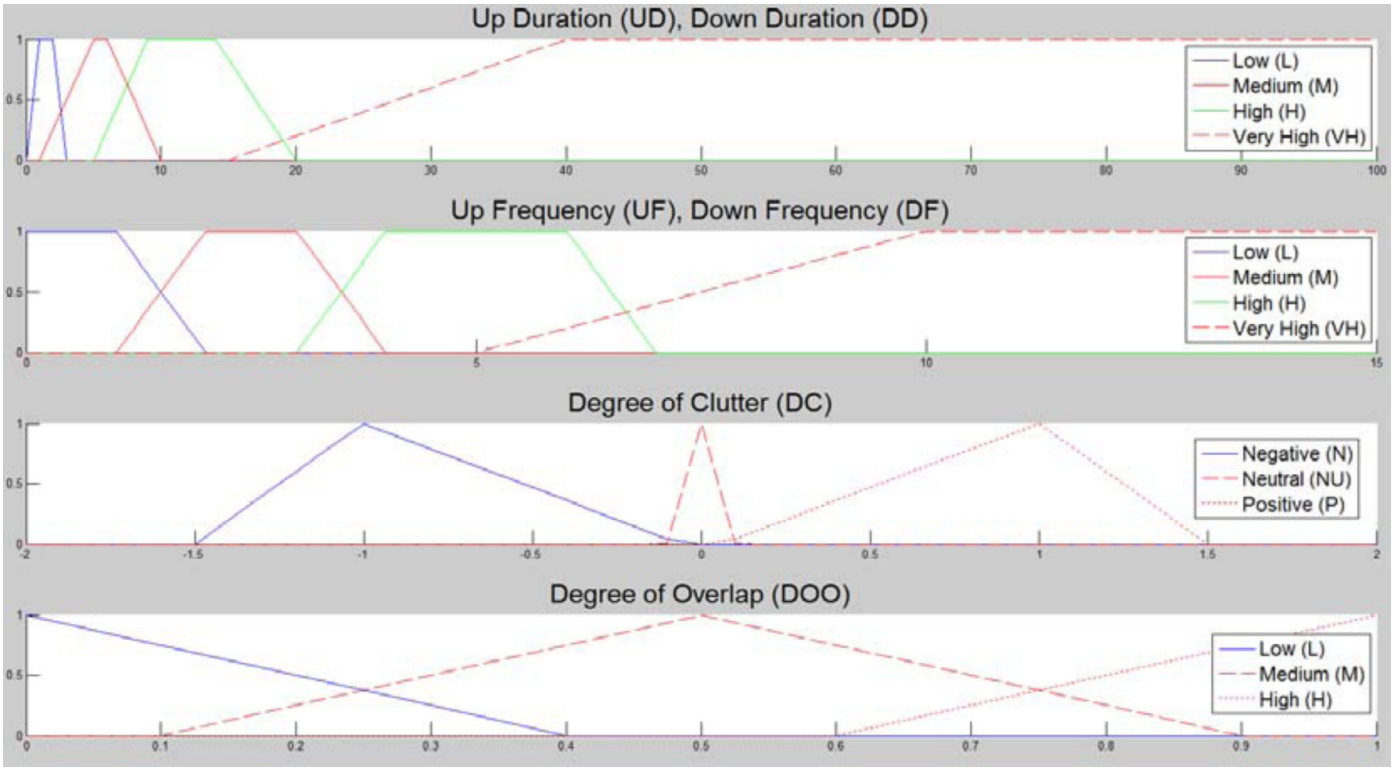

Fig. 10. Membership function plot for the second state input features.

Table 11
Set of fuzzy rules for *clean object*.

	DC	DD	DF	UD	UF	Clean object
1	P	H		H		H
2	P	VH		H		H
3	P	H		VH		H
4	P		H		H	H
5	P		VH		H	H
6	P		H		VH	H
7	P	H	H			H
8	P	VH	H			H
9	P	H	VH			H

Table 12
Set of fuzzy rules for *clutter object*.

	DC	DD	DF	UD	UF	Clutter object
1	N	H		H		H
2	N	VH		H		H
3	N	H		VH		H
4	N		H		H	H
5	N		VH		H	H
6	N		H		VH	H
7	N	H	H			H
8	N	VH	H			H
9	N	H	VH			H

that while there is interaction with an object, there is no significant change in its surface clutter during the activity. If there is any upward or downward movement near the object, the high-level activity state is *vertical movement near object*. This state is unaffected by the degree of clutter feature (DC) and is just used for any state that involves either bending or rising (or both) near an object.

Consider the high-level activity *clutter object*. This occurs when there is interaction with the object and simultaneous increase in the DC feature when we compare the surface before and after the event. At the same time, there is bending and rising movement (vertical movement) detected near that object. The object can be a bed, a table, a countertop, a couch or any other object with a horizontal surface. Using the summaries generated by the previous stage, the upward motion frequency, the upward motion duration, the downward motion frequency, and the downward motion duration are computed for every minute with a window size of 5 min.

The rules for *clean object* activity state are given in Table 11. Here, the input feature down duration (DD) is the activity summary for the state *bend near object* extracted from the first level of our FIS system. Similarly, up duration (UD) indicates the summary for *rise near object*. The membership parameters for DD and UD, as well as up frequency (UF) and down frequency (DF) are shown in Fig. 10. The output membership functions for *clean object*, *clutter object*, *rearrange object* and *vertical movement near object* are the same as in Fig. 8.

Table 13
Set of fuzzy rules for *rearrange object*.

	DC	DOO	DD	DF	UD	UF	Rearrange object
1	NU	L	H		H		H
2	NU	L	VH		H		H
3	NU	L	H		VH		H
4	NU	L		H		H	H
5	NU	L		VH		H	H
6	NU	L		H		VH	H
7	N	L	H	H			H
8	N	L	VH	H			H
9	N	L	H	VH			H

The *clutter object* fuzzy rules are given in Table 12. The only difference in the rules for *vertical movement near object* is that there is no input feature DC in the rule base. The other membership function values remain the same.

Table 13 gives the set of rules for *rearrange object* activity model. Here, the surface of the object is rearranged without any change in the degree of clutter. This activity could be a part of housekeeping

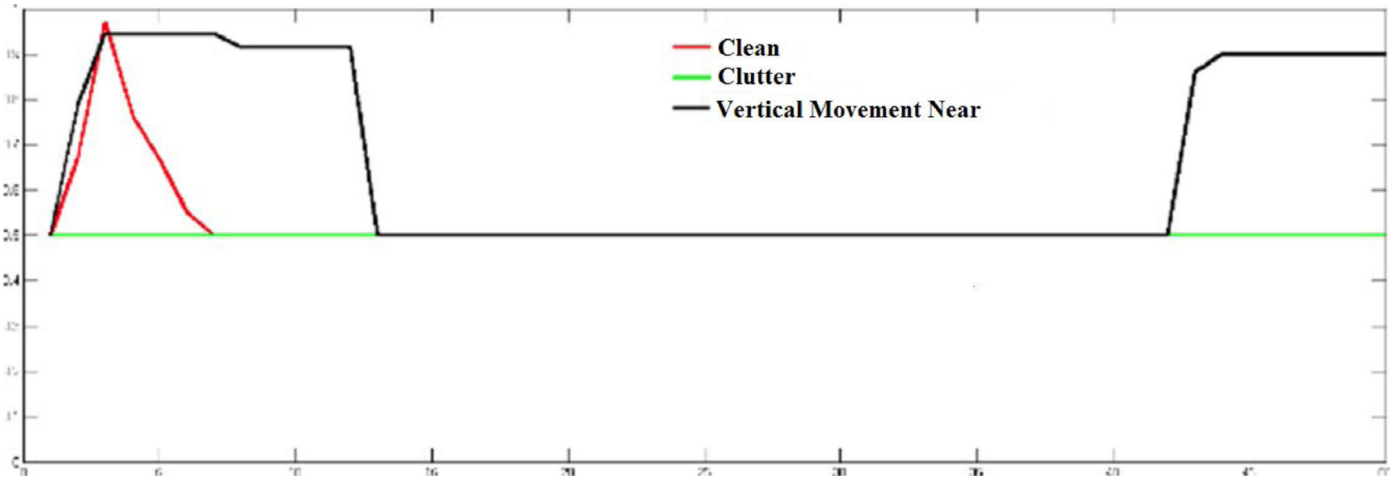

Fig. 11. High-level state membership plot for a part of the *clean object* activity state. The duration of the plot is around 5 min. Here, the *X* axis is the frame number and the *Y* axis is the confidence value for the activity states ranging from 0 to 1. (For interpretation of the references to colour in the text, the reader is referred to the web version of this article.)

when someone lifts items from a surface, cleans the items and then replaces them in a different location.

Once the confidences for the three high-level activity events are generated, summaries are created by merging two or more events which take place within a 1 min interval. We again threshold it by filtering out the states with confidence value below 0.5 to reduce the number of rules required for each activity state since it removes the need to define rules with Low output membership. The summaries for this high-level activity include the time stamps of the beginning and end of each activity segment, as well as the overall confidence of the high-level activity state is computed using the maximum confidence value generated during the event. A sample of the state memberships for a small segment of *clean object* activity is shown in Fig. 11. The red line represents the *clean object* state, the green line represents the membership for the *clutter object* state, and the black line represents the *vertical movement near object* state. We see that there is a high confidence for *vertical movement near object* initially while there is a concurrent increase in *clean object* confidence. During this entire time period, the confidence *clutter object* activity state is never above 0.5.

7.3. Third layer FIS

This is the final stage of our hierarchical fuzzy rule based system. The third stage is included to eliminate certain false alarms at the second level. One of the examples for such a scenario is the *eating* activity. Suppose a person is detected in the kitchen with multiple movements near the countertop while he is preparing the meal or snack. After that, he sits down in a chair and there is detected interaction with a table (dining table) in front while he is eating. In this case, since the person was detected with initial interaction with the countertop followed by interaction with the dining table, our confidence in the *eating* activity will be much higher than if we were just to detect the person sitting at the dining table. To build such a summary, we need at least a three layered hierarchical model. By adding this third layer, we can incorporate the events occurring before (or after) a specific event to give it more context and to generate a hierarchical order of events to define complex activities. The temporal aspect of our activity model exists in each layer of our hierarchical model. In the lowest level, the temporal aspect is present since we detect the "change" from the previous frame so some of the features such as DH are temporal in nature. For the next layer, the temporal summaries (since they contain time stamps to represent the duration of each atomic activity state) from the first layer are used as input features. Even in this final layer of our model, we use the temporal aggregated features like near table duration (NTD) that utilizes the time duration of the high level activity state. The membership functions for the three input features: vertical movement near counter duration (VMNCD), on chair duration (OCD), near table duration (NTD), and the *eating* output are shown in Fig. 12.

Table 14
Fuzzy rule base for eat activity.

	VMNCD	OCD	NTD	Eat
1	H	H	H	H
2	H	VH	H	H
3	VH	H	H	H
4	H	H	VH	H
5	H	VH	VH	H
6	VH	H	VH	H
7	VH	H	M	H
8	H	H	M	H
9	VH	M	H	H
10	H	M	H	H
11	M	H	M	H
12	M	H	M	H
13	M	H	H	H
14	M	H	VH	H
15	VH	M	H	H
16	H	M	H	H

The time stamp of the beginning of the first activity segment with a membership of over 0.5 defines the beginning of the *eating* activity segment. Correspondingly, the end of the segment is taken as the end of the activity. The maximum confidence during this interval is chosen to be the overall confidence of this detected activity. The combination of these three parameters then forms the overall *eating* activity summary. Table 14 describes the rules for the *eating* activity. An example of one of the rules is: **IF** *the activity state* ***Vertical Movement Near Counter Duration (VMNCD)*** *is* ***High*** *and* ***the On Chair Duration (OCD)*** *is* ***High*** *and the* ***Near Table Duration (NTD)*** *is* ***High***, **THEN** *the membership for* ***Eat*** *is* ***High***.

But what if the person prepares the meal at the counter and instead of sitting and eating at the dining table, chooses to eat at the counter? There could be other variations in the order of the different high-level activity states depending on the individual and other extraneous factors. The rules from Table 14 may not get fired in this case! In order to address this, we have the unknown event recognition framework described in Sections 7.1 and 7.2. In that case, although we may not be able to identify the activity state *eating*, we will still be able to get the interaction with the object surfaces (in this case, the counter and/or the dining table). This partial summary; along with some more

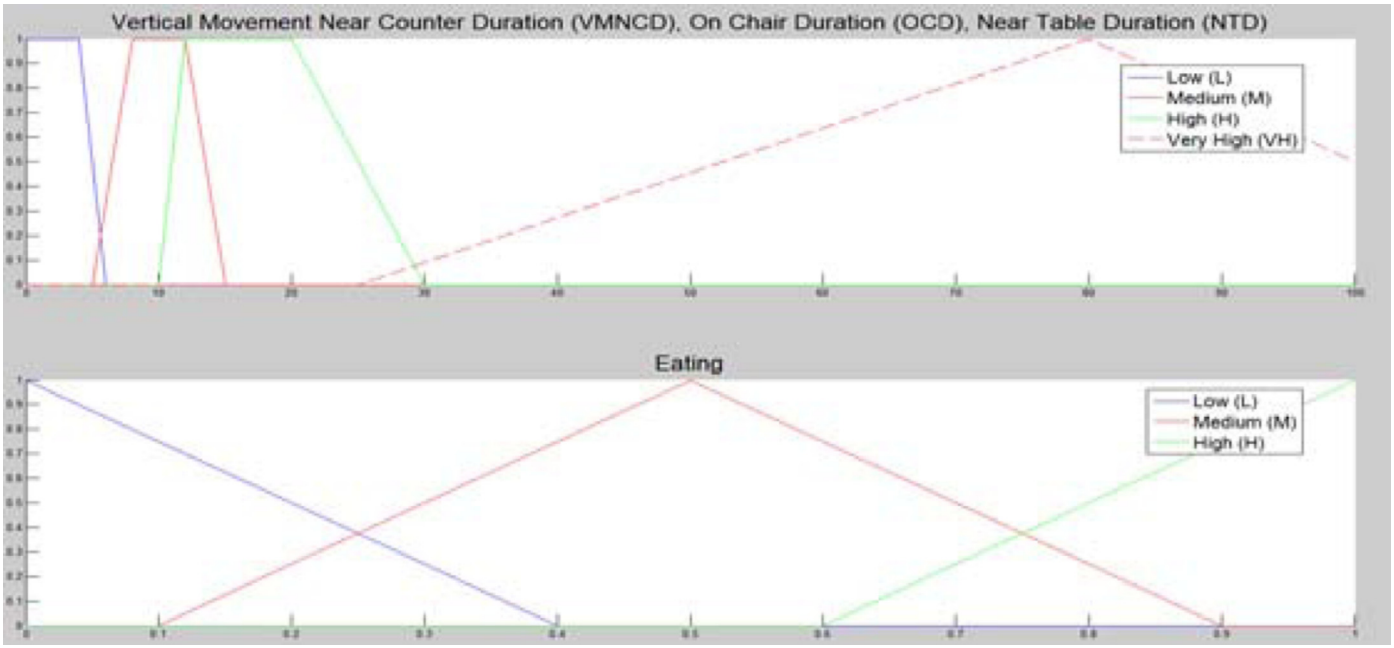

Fig. 12. Membership function plot for the second state input features VMNCD, OCD, and NTD.

contextual information from the ontology in Fig. 1 such as time of day, location in the house; can then be used for comparison with similar activity summaries.

8. Results

To test the proposed recognition methods, we use two data sets and compare results with the HMM. The first dataset was recorded in a controlled environment with subjects performing specified activities. The second is a sample of data collected in an apartment at TigerPlace with an older resident performing activities as a part of his normal routine. There are several available datasets with RGB-D data of different activities. Specifically for ADLs, the Cornell Activity Datasets: CAD 60 and CAD 120 datasets consist of several different ADLs and IADLS like eating, drinking, cleaning [31]. However, there is no contextual temporal information in the depth data. For example, the video sequence showing the eating activity just shows a person munching on an apple. There is no information about what happened before (e.g., the person went to the table and got an apple) and what happened later (e.g., cleaning up). Since our algorithm relies on this information, these datasets are unsuitable. Furthermore, the algorithm implemented by the Cornell group uses the RGB information as well as the depth data which causes privacy concerns for a continuous monitoring application.

8.1. Experiments in a controlled setting

We test our algorithm in laboratory settings with six individuals. Each individual is asked to perform the activity in a natural, normal way. Depth data are recorded using the Microsoft Kinect sensors at a frame rate of approx. 6.5 frames/s for a duration of 15 days continuously, i.e. for a period of approximately 360 h. The training data are not a part of the test sequences described in this section. There was also partial occlusion present in some of the events to test the robustness of our algorithm. The data can be downloaded from this link.[3] Healthy participants (three male and three female, height ranging from 5 ft to 6 ft, weight ranging from 115 pounds to 200 pounds) were recruited between the ages of 25–38 for this part of the experiment. They were asked to perform the activities at their normal pace, as well as at slower speeds to emulate the behavior of older adults. We then run the HMM algorithm on the same dataset with the same input features to compare the results.

An HMM is a generative probabilistic model, that generates hidden states from observable data [17]. Each activity category, such as walking, has a separate HMM that is trained via the Baum–Welch procedure [20,21]. The most likely model is calculated for each observation sequence according to the forward–backward procedure [20,21]. Since the model measures the joint likelihood of the observation sequences, the final likelihood value is very low. To rescale this value, we use the log likelihood value instead of just the probability value. The model with the highest log-likelihood value is then selected as the most likely activity label [27]. For our implementation, we use the Kevin Murphy toolbox [19] and train activity models for the following general activities that are part of ADLs and IADLs: *sitting*, *walking*, *cleaning objects*, *cluttering objects*, *moving objects*, *moving near objects*, and *rearranging objects* (described in Sections 6 and 7). The objects used are two chairs, one small table, one large table, a couch, and a bed. We also build models for two popular complex IADLs: *making bed*, and *eating a meal*. A 10-fold cross validation method was used to train and test the HMM with the number of hidden states varying from 2 to 8. The optimal number of hidden states was found to be 7, which was used for the results shown in Table 15. We implemented the Hierarchical HMM model described in [43] for the high-level activity state recognition. For the activities *clean*, *clutter* and *rearrange*, we use a two level HHMM; for *make bed* and *eat*, we use a three level HHMM. We do this to enable a fair comparison of our hierarchical algorithm with the HHMM algorithm. For our experiments, we have kept the number of substates in each level at the default value of 7. We have further implemented the algorithm described in [44] that describes a thresholded version of the HMM (HMMth) to account for the possibility that the activity may not be a part of any of the categories. We do so by creating a weak classifier that trains on all the activities and provides the minimum threshold to decide if the current activity belongs to the categories or not [44].

From Table 15, we see that our algorithm outperforms the HMM model. The results highlight the strengths of our proposed algorithm.

[3] http://www.eldertech.missouri.edu/adl/

Table 15
Activity detection (AD) and false positive (FP) results using lab dataset. Here HMM refers to HMM for the atomic activity states and HHMM for the complex activities. HMMth refers to the thresholded version of HMM described earlier in the section.

Activities	HMM AD	HMM FP	HMMth AD	HMMth FP	FIS AD	FIS FP
Walk	47/50	17	47/50	11	50/50	2
Sit	46/50	13	46/50	8	50/50	5
Clean object	26/30	13	26/30	11	28/30	6
Clutter object	26/30	13	25/30	13	29/30	5
Move near object	27/30	16	27/30	14	30/30	2
Rearrange object	25/30	14	24/30	12	28/30	5
Move object	26/30	11	26/30	11	29/30	5
Make bed	20/24	6	18/24	4	24/24	3
Eat	17/20	9	16/20	8	19/20	4

Fig. 13. The system setup at TigerPlace. The Kinect sensor is positioned over the front door of the apartment and the computer is placed inside the cabinet highlighted in yellow. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

The one instance that our algorithm was not able to detect the *eating* activity was due to the subject being outside the field of view while preparing the meal. This generated a low confidence for the activity state *vertical movement near object* (in this case the counter) which consequently affected the confidence for the *eating* activity state. For the false positives, the HMMth significantly improves the performance for the HMM approach as compared to the baseline HMM method. However, in some cases it also slightly reduces the recognition rate, e.g. for the *clutter object* activity. The overall better performance of our model is not surprising since our system can handle uncertainty in the activity due to its fuzzy rule based system. For example, in our data set for the controlled setting (Section 8.1), for one of the instances, Actor 2 went to the table a couple of times to get water. On the other hand, Actor 1 did not get water while eating. In the training data for our fuzzy model, the actor (who was not a part of the controlled setting experiment) did not get up and fetch water during eating. However, in both the cases, the activity was identified as eating, although for different durations and with different confidences. In the former case, the confidence was 0.8 as compared to the second instance (0.72) since the vertical movement near counter duration (VMNCD) parameter was higher in the former case. Thus, our model can incorporate more variations in a given ADL/IADL model than other activity models.

8.2. Experiments in a home environment collected at TigerPlace

In this section, we evaluate the performance of the algorithm from depth data collected in an apartment of a resident at TigerPlace. The resident is of age 88, without any cognitive impairments, living independently at TigerPlace. He does not use any ambulatory support such as a walker, and suffers from muscle weakness, and chronic conditions like asthma and diabetes. He currently does not need any support from the TigerPlace staff to perform IADLS and ADLs which makes him an ideal subject to test our system. The system consists of a single Microsoft Kinect sensor mounted near the ceiling above the front door of the apartment and a computer placed in a cabinet above the refrigerator (Fig. 13). The data are collected in an unstructured setting while the elderly resident is going through his normal routine. Six hours of continuous depth data are processed and analyzed to evaluate the performance of our algorithm. The data are captured at approximately 6.5 frames/s using the same data capture process as for the controlled environment described in Section 8.1. Since this is an unscripted data capture in an unstructured environment, there are several instances of occlusion present which further test the robustness of our algorithm. During this time, there are several instances of sitting, walking, and moving around in the room as the resident goes through his normal routine. There are two *eating* activities performed during this time period.

Fig. 14 shows the images of the resident preparing his meal at the counter (a) and eating the meal at the dining table (c). The locations are labeled for clarity.

The results of the activity detection from the TigerPlace dataset are shown in Table 16. The FIS system and the HMM models used in this experiment are the same as those described in Section 8.1. The data are manually labeled to obtain the ground truth.

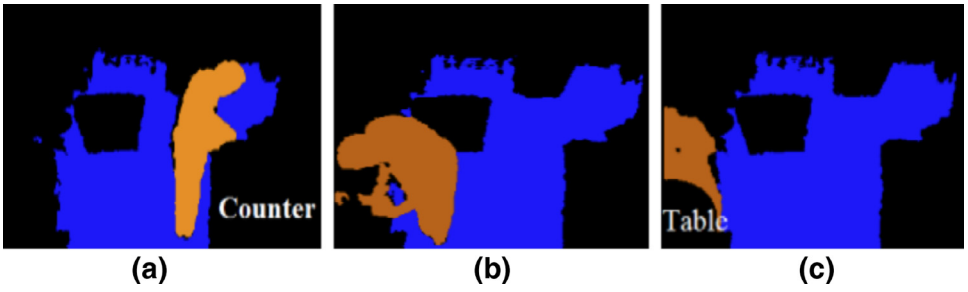

Fig. 14. Foreground images of eating activity at TigerPlace. Here, the resident (a) prepares the meal on the countertop, (b) places the items of the meal on the dining table and (c) finally sits down and eats the meal at the table.

Table 16
Activity detection results using HMM and FIS.

Activities	HMM AD	HMM FP	HHMth AD	HMMth FP	FIS AD	FIS FP
Walk	14/15	6	14/15	5	15/15	0
Sit	5/8	5	5/8	3	7/8	0
Clean object	3/4	7	3/4	4	4/4	1
Clutter object	2/3	4	2/3	4	3/3	1
Move near object	5/7	3	4/7	3	7/7	0
Rearrange object	2/2	6	2/2	5	2/2	1
Eat	1/2	1	1/2	0	2/2	0

The results show that the FIS outperforms the both the HMM models in the unstructured settings. Similar to the results obtained in Section 8.1, the HMMth has a lower false alarm rate, although in the case of the *move near object* high-level activity state, the performance decreases slightly due to the thresholding effect of the algorithm. This further highlights the robust performance of our algorithm in dynamic settings. The one instance where our algorithm fails to detect the person sitting is due to the chair being located far away from the sensor. The chair is located at a distance greater than 6 m from the location of the Kinect. This exceeds the range and is one of the limitations of the Kinect depth sensor. The false positives generated by our algorithm for the *clean object*, *clutter object*, and *rearrange object* high-level activity states indicate that there is definitely room for improvement in our approach that may be eliminated by incorporating machine learning methods to "learn" the parameter values of the fuzzy rules. However, our approach still outperforms the HMM and its variation approaches in the unconstrained environment that highlight its generalizability and emphasize its potential to be used in a continuous monitoring application to detect health change from behavior patterns.

9. Conclusions

We demonstrate a flexible framework for detecting ADLs in an in-home environment using depth data from the Kinect sensor. Depth data provide the added advantage of unobtrusive monitoring with its ability to perform just as well under different lighting conditions. Silhouette features from the depth data as well as scene features are extracted and input to a fuzzy inference system, and activity states of the individuals are determined using fuzzy confidence measures. The resulting fuzzy rule based outputs are then temporally processed and used to generate temporal activity summaries. These summaries are input to another fuzzy inference system for further activity reasoning. This approach results in human understandable information and confidences regarding activities which can be used to monitor the activity patterns of older adults in their daily routine. The generalized framework can handle uncertainties in activities performed and generate useful information even with automatically extracted unlabeled object surfaces. In addition to the lab experiments, the algorithm has been tested using data collected at TigerPlace and has shown its robustness in unstructured, dynamic environments.

10. Future work

We highlight the importance of our algorithm's generalized framework for activity modeling. One question that needs answering is how is this useful in a dynamic unstructured environment where there are some unlabeled surfaces present? Since our algorithm updates the surfaces in the field of view on a daily basis as well as when there is a detected change in any object surface, it can identify new surfaces that are yet to be labeled. Our algorithm then identifies the occurring activity as: There is interaction with Surface No. 2 at **X** time (time of day) for **Y** duration. We can thus get useful information about an unknown activity. The next question then is how is this useful for monitoring day to day behavior patterns? We believe that these interactions can be analyzed over some time period and trends from these interactions can be detected. For example, there could be interaction with Surface No. 2 at **X** time every day. The interactions with all the surfaces can be used to study behavior patterns and following that, to detect anomalies in behavior over extended time periods, e.g. if the interaction was missing for a few days, that would indicate a change from the normal behavior pattern.

Many of the quantities used in this work are based on empirical observations. To address this, automated tuning of rules and membership functions using evolutionary computation techniques will also be considered. Currently, we have manually labeled the horizontal surfaces with their labels for instantiations, such as the bed surface; as well as the locations such as bedroom, living areas. In the future, we hope to automatically identify the objects by incorporating object recognition techniques and using context to glean the location information (e.g. bed is most likely to be in the bedroom). We have currently looked at activities involving interaction with only horizontal surfaces. We plan to extend this work to vertical as well as oblique surfaces using the HONV feature information. This will further improve the generalizability of our algorithm. Also, we have used only depth data in our current approach. We plan on testing features extracted from both, the color data as well as the depth data and compare the systems.

We build a framework that generates temporal activity summaries of daily behavior over longer time periods. This can be coupled with activity information obtained from other sensors such as bed sensors, acoustic sensors, and motion sensors to generate more descriptive activity patterns. An in-home activity monitoring system would benefit greatly from our algorithm to alert healthcare providers of significant temporal changes in ADL behavior patterns of frail older adults for fall risk and cognitive impairment.

Acknowledgments

This work was supported in part by the Agency for Healthcare Research and Quality under grant R01-HS018477 and NSF grant CNS-0931607. The authors would like to thank members of the Eldercare and Rehabilitation Technology team for their support.

References

[1] D.G. Lowe. Object recognition from local scale-invariant features, in: Proc. ICCV, Kerkyra, Greece, 1999, pp. 1150–1157.
[2] E.E. Stone, M. Skubic, Unobtrusive, continuous, in-home gait measurement using the Microsoft Kinect, IEEE Trans. Biomed. Eng. 60 (10) (2013) 2925–2932.
[3] A. Telea, An image inpainting technique based on the fast marching method, J. *Graphics Tools* 9 (1) (2004) 23–34.
[4] F. Clark, S.P. Azen, M. Carlson, D. Mandel, L. LaBree, J. Hay, R. Zemke, J. Jackson, L. Lipson, Embedding health promoting changes into the daily lives of independent-living older adults, J. Gerontol. Ser. B, Psychol. Sci. Soc. Sci. 56 (2001) 60–63.
[5] A. Zisberg, H.M. Young, K. Schepp, Development and psychometric testing of the Scale of Older Adults' Routine, J. Adv. Nurs. 65 (3) (2009) 672–683.
[6] L. Chen, C.D. Nugent, H. Wang, A knowledge-driven approach to activity recognition in smart homes, IEEE Trans. Knowl. Data Eng. 24 (6) (2012) 961–974.
[7] F. Latfi, B. Lefebvre, C. Descheneaux, Ontology-based management of the Tele-Health smart home, dedicated to elderly in loss of cognitive autonomy, in: CEUR Workshop Proceedings, 258, 2007.

[8] M.D. Rodríguez, M. Tentori, J. Favela, D. Saldaña-Jimenez, J. García-Vázquez, CARe: an ontology for representing context of activity–aware healthcare environments, in: Activity Context Representation, 2011.
[9] C. Graf, The Lawton instrumental Activities of Daily Living (IADL) scale., Medsurg. Nurs. 18 (2009) 315–316.
[10] D. Anderson, R.H. Luke, J.M. Keller, M. Skubic, M. Rantz, M. Aud, Linguistic summarization of video for fall detection using Voxel Person and fuzzy logic, Comput. Vis. Image Understanding 113 (January (1)) (2009) 80–89.
[11] M. Shelkey, M. Wallace, Katz index of independence in activities of daily living (ADL), Gerontologist 185 (2) (1998).
[12] T. Banerjee, S. Member, J.M. Keller, M. Skubic, Resident identification using Kinect depth image data and fuzzy clustering techniques, in: International Conference of the IEEE Engineering in Medicine and Biology Society, 2012.
[13] S. Tang, X. Wang, X. Lv, T.X. Han, J. Keller, Z. He, M. Skubic, S. Lao, Histogram of oriented normal vectors for object recognition with a depth sensor, in: The 11th Asian Conference on Computer Vision (ACCV), 2012.
[14] C. Liu, J. Yuen, A. Torralba, SIFT flow: dense correspondence across different scenes and its applications, IEEE Trans. Pattern Anal. Mach. Intell. 33 (5) (2011).
[15] B. Oehler, J. Stuckler, J. Welle, D. Schulz, S. Behnke, Efficient multi-resolution plane segmentation of 3D point clouds, in: Proc. of the International Conference on Intelligent Robotics and Applications (ICIRA), Aachen, Germany, 2011, pp. 145–156.
[16] T. Banerjee, J.M. Keller, M. Skubic, Detecting foreground disambiguation of depth images using fuzzy logic, in: 2013 IEEE International Conference on Fuzzy System, pp. 1–7, July 2013.
[17] C. Sutton, A. McCallum, L. Getoor, B. Taskar, An introduction to conditional random fields for relational learning, Introduction to Statistical Relational Learning, MIT Press, 2006.
[18] P.-C. Chung, C.-D. Liu, A daily behavior enabled hidden Markov model for human behavior understanding, Pattern Recognit 41 (May (5)) (2008) 1572–1580.
[19] K. Murphy, The Bayes net toolbox for MATLAB, Comput. Sci. Stat. 33 (2001) 1024–1034.
[20] L. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, Proc. IEEE 77 (February (2)) (1989) 257–286.
[21] J. Bilmes, A Gentle Tutorial on the EM Algorithm and Its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models, University of Berkley, 1988 , Technical report ICSI-TR-97-021.
[22] D. Cook, N. Krishnan, Z. Wemlinger, Learning a taxonomy of predefined and discovered activity patterns, J. Ambient Intell. Smart Environ. 5 (6) (2013) 621–637.
[23] E. Nazerfard, D. Cook, Using Bayesian networks for daily activity prediction, in: Twenty-Seventh AAAI, 2013.
[24] G. Lavee, E. Rivlin, M. Rudzsky, Understanding video events: a survey of methods for automatic interpretation of semantic occurrences in video, IEEE Syst. Man Cybern. 39 (September (5)) (2009) 489–504.
[25] N.T. Nguyen, D.Q. Phung, S. Venkatesh, H. Bui, Learning and detecting activities from movement trajectories using the hierarchical hidden Markov models, in: IEEE Computer Society Conference on Computer Vision Pattern Recognition, Washington, 2005, pp. 955–960.
[26] T.V. Duong, H.H. Bui, D.Q. Phung, S. Venkatesh, Activity recognition and abnormality detection with the switching hidden semi-Markov model, Proc. *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* 1 (2005) 838–845.
[27] S. Gong, H. Buxton, On the visual expectations of moving objects, in: Proceedings of the 10th *European Conference on Artificial Intelligence (ECAI)*, Wiley, New York, 1992, pp. 781–784.
[28] L. Zadeh, Fuzzy sets, Inf. Control 8 (3) (1965) 338–353.
[29] A. Zadeh, Outline of a new approach to the analysis of complex systems and decision processes, IEEE Trans. Syst. Man Cybern. SMC-3 (1973) 28–44.
[30] E. H. Mamdani, S. Assilian, An experiment in linguistic synthesis with a fuzzy logic controller, Int. J. Man-Mach. Stud. 7 (1) (1975) 1–13.
[31] J. Sung, C. Ponce, B. Selman, A. Saxena, Unstructured human activity detection from rgbd images, in: ICRA, 2012.
[32] D. Brulin, Y. Benezeth, E. Courtial, Posture recognition based on fuzzy logic for home monitoring of the elderly, IEEE Trans. Inf. Technol. Biomed. 16 (5) (2012) 974–982.
[33] L.E. Parker, 4-Dimensional local spatio-temporal features for human activity recognition, in: IEEE/RSJ International Conference on Intelligent Robotic System, pp. 2044–2049, September 2011.
[34] C. Peters, T. Hermann, S. Wachsmuth, J. Hoey, Automatic task assistance for people with cognitive disabilities in brushing teeth—a user study with the TEBRA system, ACM Trans. Access. Comput. 5 (March (4)) (2014) 1–34.
[35] H. Pirsiavash, D. Ramanan, Detecting activities of daily living in first-person camera views, IEEE Conf. Comput. Vis. Pattern Recognit. (June) (2012) 2847–2854.
[36] H.S. Koppula, R. Gupta, A. Saxena, Learning human activities and object affordances from RGB-D videos, Int. J. Robot. Res. 32 (July (8)) (2013) 951–970.
[37] M. Bengalur, Human activity recognition using body pose features and support vector machine, in: International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp.1970-1975, 22–25 August 2013.
[38] C. Zhang, Y. Tian, RGB-D camera-based daily living activity recognition, J. Comput. Vis. Image Process. 2 (December (4)) (2012).
[39] G. Demiris, O. Parker, J. Giger, M. Skubic, M. Rantz, Older adults' privacy considerations for vision based recognition methods of eldercare applications, Technol. Health Care 17 (1) (2009) 41–48.
[40] G. Meditskos, S. Dasiopoulou, V. Efstathiou, I. Kompatsiaris, Ontology Patterns for Complex Activity Modelling Theory, Practice, and Applications of Rules on the Web, Lecture Notes in Computer Science 8035 (2013) 144–157.
[41] L. Chen, C. Nugent, Ontology-based activity recognition in intelligent pervasive environments, Int. J. Web Inf. Syst. 5 (4) (2009) 410–430.
[42] D. Saldana-Jimenez, M. Rodriguez, J. Garcia-Vazquez, A. Espinoza, Elder: an ontology for enabling living independently of risks, OTM 2009 Workshops, 5872, LNCS, 2009, pp. 622–627.
[43] S. Fine, Y. Singer, N. Tishby, The hierarchical hidden Markov model: analysis and applications, Mach. Learn. 62 (1998) 41–62.
[44] H. Lee, J. Kim, An HMM-based threshold model approach for gesture recognition, IEEE Trans. Pattern Anal. Mach. Intell. 21 (October (10)) (1999) 961–973.
[45] K. Khoshelham, S. Elberink, Accuracy and resolution of Kinect depth data for indoor mapping applications, Sensors 12 (2) (2012) 1437–1454.