

# Ontology-driven semantic unified modelling for concurrent activity recognition (OSCAR)

Muhammad Safyan<sup>1,2</sup> · Zia Ul Qayyum<sup>2</sup> ·  
Sohail Sarwar<sup>2</sup> · Raúl García-Castro<sup>3</sup> · Mehtab Ahmed<sup>4</sup>

Received: 11 April 2017 / Revised: 19 June 2018 / Accepted: 25 June 2018 /  
Published online: 5 July 2018  
© Springer Science+Business Media, LLC, part of Springer Nature 2018

**Abstract** Activity recognition has a vital role in smart home operations. One of the major challenges in object-sensor-based activity recognition is to learn the complete activity model derived from a generic activity model for sequential and parallel activities. Such challenge exists due to erratic degrees of dissimilar activities in which inhabitants perform activities in sequential and interleaved fashion while interacting with different objects. The proposed work focuses on recognizing a complete set of actions (of activity) by exploiting different knowledge engineering techniques, ontology-based temporal formalisms and data driven techniques. Semantic Segmentation has been employed to establish the generic activity model. The spurious semantic segmentation produced by sensor noise or erratic behaviour is removed by Allen’s temporal formalism. Moreover, Tversky’s feature-based similarity has been used to remove the highly similar spurious activities produced as a result of mistaken interactions with wrong home objects. The duration to perform activities varies among inhabitants; such duration intervals are identified dynamically using the proposed model in order to have a complete activity model. A comprehensive set of experiments has been carried out for evaluating the proposed model where the results based upon different metrics assert its effectiveness especially when compared with other contemporary techniques.

**Keywords** Complete activity model · Personalized activity model · Adaptive system · Domain activity ontology · Concurrent activity recognition

---

**Electronic supplementary material** The online version of this article (<https://doi.org/10.1007/s11042-018-6318-5>) contains supplementary material, which is available to authorized users.

---

✉ Muhammad Safyan  
m.safyan@seecs.edu.pk

<sup>1</sup> Iqra University Islamabad, Islamabad, Pakistan

<sup>2</sup> University of Gujarat, Gujarat, Pakistan

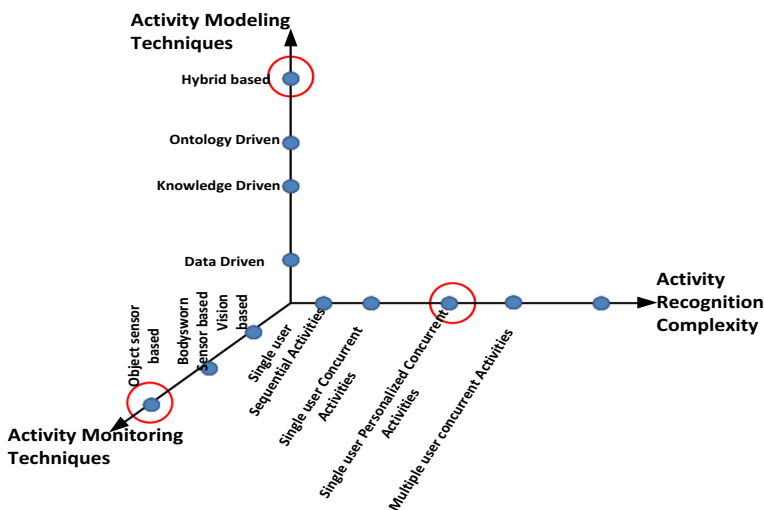
<sup>3</sup> Ontology Engineering Group, Universidad Politécnica de Madrid, Madrid, Spain

<sup>4</sup> GC University, Lahore, Pakistan

## 1 Introduction

The internet of Things (IoT) has been a success since its inception [18]; the intention is to make human life comfortable in different ways such as smart homes [7, 18] and smart cities [5]. “Activity Recognition” is an important aspect to build such facilities especially in smart homes. It transforms a home into a smart home for the inhabitants (focus of our work). Activity Recognition (AR) process, as illustrated in Fig. 1, can be comprehended from three perspectives (i) Activity Monitoring (ii) Activity Modeling and (iii) Activity Complexity. Each of the perspectives is elaborated as given in the following.

Activity monitoring describes the process of capturing actions performed in pervasive environments. Some of the prevalent AR techniques are: *Vision-based AR*: An image-based approach that involves monitoring of inhabitant’s actions, activities and their complete behaviour using surveillance cameras. Based on the major features of a vision-based approach, it identifies the events by discovering areas of interest, observing motion patterns and features like walking, hand waving or running [15, 17, 37]. On the other hand, these techniques compromise the inhabitant’s privacy, use higher bandwidth and add a cost of computational resources [41]. *Sensor-based AR*: In order to identify the events taking place among inhabitants, sensor streams are received from sensors [8]. The sensor-based AR techniques have two sub-categories: *Wearable-sensor-based AR*: Wearable sensors are attached to the parts of inhabitant’s body for capturing kinesis. Activities are identified by data mining or machine learning techniques using the data captured through sensors. The major challenge of these techniques is the identification of complex and similar tasks [40]; for example, “making coffee” and “making tea”. The usage of wearable sensors is limited due to constraints of sensor size, battery life, and reluctance of inhabitants in wearing sensors [8]. *Object-sensors-based AR*: Sensors generate diverse signals as a result of interaction with home objects such as “cups”, “jars”, “cutlery” etc. [4, 8], and this collection of sensor values (or signals) is used for the recognition of activities [8]. Object-based activity recognition can be a potential success due to its



**Fig. 1** Activity recognition perspectives

affordability, low power consumption, real-time response and perfect measurement [7]. In this work, the aspect of *object-sensor-based AR* is employed due to its viability and effectiveness in smart homes [40].

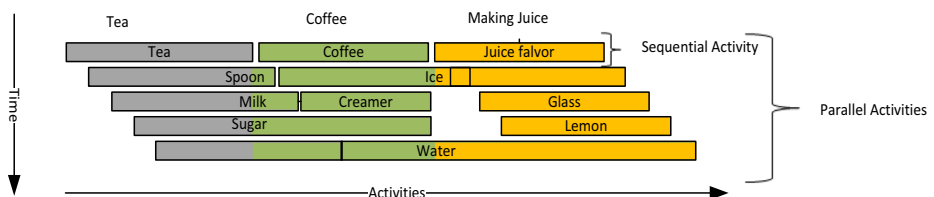
A second perspective of activity recognition is the modelling of sensor data streams that can be carried out by three techniques: *Data-driven Techniques* are based upon machine learning and data mining approaches [4], which are widely used in AR processes; these techniques are signified by their ability to handle noise, uncertainty, and incompleteness in sensor data streams [12]. On the other hand, they pose different challenges such as: cold start or data scarcity [8], non-reusable/non-scalable, complex model, and high computational costs [13]. *Knowledge-driven Techniques*: exploit prior knowledge that is expressive and rich in meaningful patterns [7]. There is a variety of knowledge-driven techniques such as: rule-based, mining-based and ontology-based [8] techniques where AR mostly focuses on ontology-based approaches. Ontology-based techniques [6] are easy to start, semantically consistent and logically elegant [42, 43]; however, these techniques have shortcomings in managing dubious actions, temporal information and incomplete models [13]. *Hybrid Techniques*: are the combination of data-driven approaches and knowledge-driven techniques for activity modelling in a real-time environment [23, 24].

The perspective of activity recognition complexity describes the intricacy of action sequences performed by inhabitants in a smart home. It is triggered by the sequential activity of a “single user” and lasts with concurrent activities of “multiple users”.

This paper focuses on exploiting both data-driven approaches and knowledge-driven techniques for concurrent activity recognition in a personalized fashion. The dimension spaces of proposed research have been encircled in Fig. 1.

In existing research, *object-sensor-based AR* has been used where three major challenges were identified in recognizing parallel activities: (i) home-objects may be used in multiple activities in a smart home. It is very difficult to recognize the context of object-sensor for which activity is performed especially for parallel activities [16]; e.g., “water” is a common object used in multiple activities such as “making pasta”, “making tea” and “making juice”, as shown in Fig. 2. Therefore, activities for which “water” is used need to be identified rightly. (ii) Unintentional (or mistaken) interaction with objects that are not part of ongoing activities are referred to as sensor noise or noisy actions [36]; it is also a major challenge to distinguish such noise in parallel activity recognition. Noisy actions may result in spurious activity; e.g., touching the “coffee jar” mistakenly in process of “making tea”. (iii) Extracting the start and end time (dynamic interval) of parallel activities is another challenge that has been addressed in this paper.

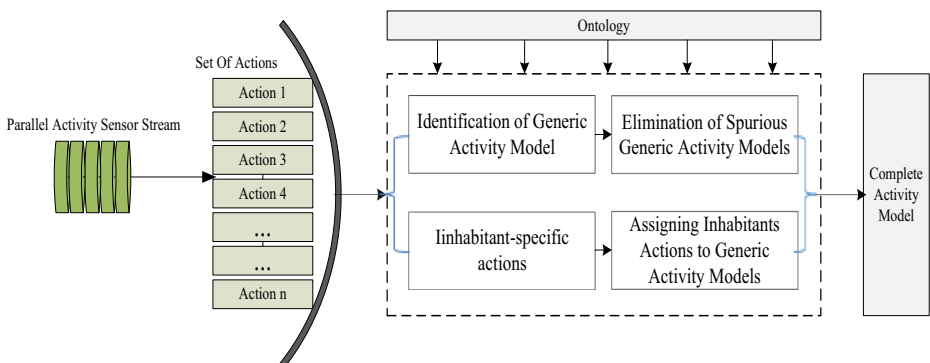
In order to address the challenges, an “Ontology-based Semantic Concurrent Activity Recognition” (OSCAR) framework is proposed based on *knowledge-driven approaches*



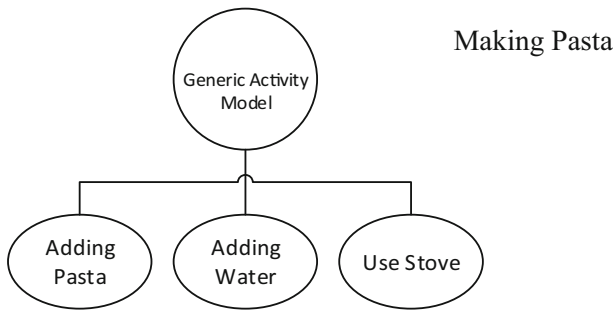
**Fig. 2** Parallel activity sensor stream in smart home

exploiting different aspects of data-driven approaches as shown in Fig. 3 (at abstract level of granularity). The proposed framework is a hybrid of knowledge-driven techniques (ontological constructs), temporal formalisms [1, 2] and data driven techniques for recognition of the complete action sequence of parallel and interleaved activities. OSCAR uses the same perceptible activity model for all inhabitants. Perceptible activity, as a model, establishes a platform for recognizing a complete activity model where activities run in parallel for inhabitants in a personalized way. The perceptible model of activities, modelled in an ontology, provides a context to recognize complete activity models of concurrent activities. The context provided by the ontology model is used in different components of OSCAR (further elaborated in section 3 and Fig. 6). The “Identification of Generic Activity Model” and “Identification of Inhabitant-specific Actions” components exploit the context of “duration”, “location” and “used-in” ontological properties to cluster the action stream. “Elimination of spurious generic activity model component” removes the spurious segments through temporal and spatial contexts. In the proposed model, the temporal context is developed by using a 4-D extended fluent approach [34, 48]. “Assigning Inhabitant-specific Action to Generic Activity Model” identifies the start and end time (dynamic duration) of ongoing activities by exploiting the “duration” property to determine the complete activity model. The context of the “used-in” property is used to calculate the similarity score used between the perceptible activity model and inhabitant-specific actions for identifying spurious activities. Lastly, the hierarchical nature of ontology [19, 25] ensures the benefit of interoperability and reusability without preparing the system over a huge dataset.

The proposed framework has a generic ontological model for each activity called Perceptible Activity Model (PAM). Each PAM contains a set of essential actions. No activity can be performed without these essential actions. The AR process takes the PAM and a stream of user-performed actions as input, attains the ontological context of PAM and produces the personalized/complete activity model for inhabitants. Complete Activity Model (CAM) of an activity is comprised of the PAM’s action plus inhabitant-specific actions. The PAM remains the same for all individuals and does not need not to be learnt each time, while the CAM may vary from time to time for an inhabitant. For example, the PAM for the “making pasta” activity, as shown in Fig. 4, has a set of essential actions: “adding pasta”, “adding water”, and “using stove”. Whereas the CAM (as



**Fig. 3** An abstract view of OSCAR

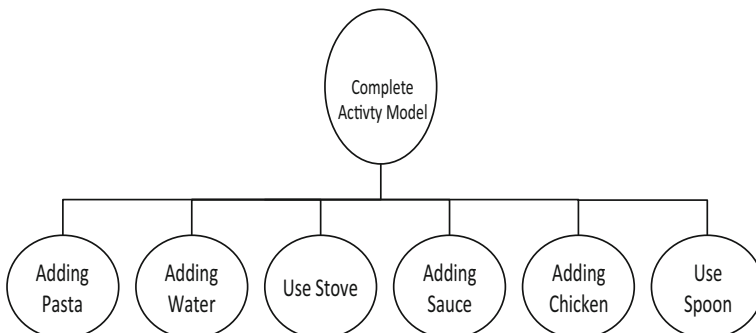


**Fig. 4** Perceptible/Generic activity model

shown in Fig. 5) for the same activity has a set of individual-specific actions such as “adding sauce”, “adding chicken” and “using spoon” along with actions of the perceptible model.

Keeping this in view, our research contributions, as elaborated in the following sections, are: (i) carrying out activity recognition through learning from personalized actions of an inhabitant by using the context of the ontological perceptible activity model (PAM). (ii) Distinguishing an intentional object interaction from a mistaken contact by examining the sensor stream. For example, inhabitants wrongly touching the “sugar jar” while “making pasta” whereas “sugar jar” is used in “making tea”. Such mal-interactions have been identified as sensor noise. (iii) Comprehending a random or variable sequence of actions performed by inhabitants. For example, the activity of “making tea” should be correctly recognised for all the inhabitants if some inhabitant adds “sugar” in the middle of the “making tea” activity while another inhabitant adds “sugar” at the end of the “making tea” activity. (iv) Learning the time interval dynamically for complete activity model. (v) Learning the context of home-object usage. The same type of objects may be used for different purposes. For example, a “cup” is used as a drinking container in tea, coffee or juice in the kitchen or it may be used in the wash room as a “shaving-cup”. The proposed system has the capacity to figure out the context by specifying interdependency among actions through temporal and spatial relations.

The rest of the paper is organized as follows. Section 2 provides the review of efforts made for improving AR systems; Section 3 elaborates the proposed architecture; Section 4 describes the dataset and evaluation methodology, while Section 5 discusses the results and discussions. Section 6 elaborates the conclusions and future directions of work.



**Fig. 5** Complete/Personalized activity model

## 2 Literature survey

Chen and colleagues [9] approached the issue of AR using a hybrid system to identify the activities where the ontological model updates its descriptive properties based upon learning the activity log file. The paper does not cater for the complete activity model of the inhabitant or the number of actions used to perform the activity. Successive activities can be recognized using the context knowledge of the ontology. Parallel activities and the process of handling the sensor noise have not been addressed in this paper; besides, a static window size has been used to segment the sensor stream instead of a dynamic one.

The work of Azkune and colleagues [3] is similar to the work proposed in this paper since it uses data-driven techniques. It uses the contextual knowledge to recognize the personalized activity model and a specialized model of an existing activity. However, this work recognizes only the sequential activities lacking the ability to recognize the parallel activities. The temporal information has not been catered which is an integral part of activity recognition rather “duration”, “location” and “activity-type” properties have been used to recognize the activities. A dynamic window size to segment the sensor stream is employed.

Okeyo and colleagues [38] combine ontological and temporal knowledge formalisms to provide a representation for composite activity modelling. This paper also describes the entailment rules so as to dynamically infer the composite activities. Simple activities modelled in this paper are static in nature. Our work is distinguished from this work in two aspects. Firstly, it generates a complete activity model over the foundation of a generic model. Secondly, it recognizes the activity intervals dynamically.

A knowledge-based approach for a concurrent AR has been presented by Ye and colleagues [49]. This approach explores the context of sensor activation and uses context dissimilarity to cluster a continuous sensor sequence into chunks. Each cluster corresponds to one “under process” activity. It exploits the Pyramid Match Kernel (PMK) approach, augmented with a WordNet matching on hierarchical concepts in order to recognize activities using the domain ontology from a potentially noisy sensor sequence. Author uses the dynamic window size to segment the sensor stream.

Meditkos and colleagues [35] propose a technique for identifying parallel activities of inhabitants using ontologies to encode the context knowledge, dependencies among the activities and de-feasible reasoning for conflict resolution. The recommended architecture has been incorporated in a subsisting context-cognizant ADL apperception framework, used to fortify the diagnosis of the dementia patients in a controlled environment.

In Helaoui and colleagues’ work [21], the difficulty of discovering parallel activities is addressed by coalescing the statistical-temporal models obtained from training data and background erudition as first-order temporal rules. While encoding strict temporal rules often fails to integrate the level of flexibility required in an ambient environment, the combination of data-driven and knowledge-driven solutions seems unfeasible. Also, the combination of data and knowledge-driven solutions appears to be promising, but the definitions of strict temporal rules often fail to incorporate the level of flexibility required in pervasive environments.

Gayathri and colleagues [14] focus on probabilistic approaches as well as on ontology-based models independently. Data obtained from sensors is uncertain in nature and mapping uncertainty over ontology will not yield good accuracy in the context of AR. The proposed system augments

ontology-based activity recognition with probabilistic reasoning through a Markov Logic Network (MLN) which is a statistical relational learning approach. The proposed framework uses the model theoretic semantic property of description logic, to change over the ontology action display in its relating first request rules. MLN is built by learning weighted first request decides that empower probabilistic thinking inside an information portrayal structure.

Riboni and colleagues' framework [44] proposes a representation of sensors, devices, activities and atomic actions. Their approach demonstrates a method for combining DL with probabilistic thinking. In any case, the likelihood esteems used are not established in the semantics and were given names manually. Also, the approach models concentration without considering portions of relations. Moreover, the proposed approach is static in nature with DL rules. The model orders the actions in the event that it may not characterize the DL. This work recognizes the activities occurrence but lacks the capacity to recognize the personalized behaviour of the inhabitants.

Liu and colleagues [30] propose a graphical model by combining the Chinese Restaurant Process model with Allen's temporal relations. The proposed model is assessed by utilizing two datasets, i.e., genuine video information and a brilliant home dataset (with normal exactness of 90%). An enhanced expectation in view of the Bayesian system was proposed for action acknowledgment. The exactness related to transient could not manage last movement display. These strategies were restricted to the particular order that they were intended for.

Cui and colleagues [11] propose a technique to incorporate the model learning approach (GPDM-APF) with standard APF into one structure. The two parallel trackers could run independently and melded by an arrangement of criteria; this affected the framework to pick the tracker which performed better as it yielded. By combining the two trackers, this framework could outperform others using a single approach. In any case, it could cost plenty of time to keep the two trackers running in parallel.

Liu and colleagues [33] highlighted a huge challenge for sensor-based activity recognition processes. The action highlights were extricated from the increasing speed information gathered by cell phones. An unsupervised characterization technique called MCODE was used for activity recognition. However, this method had a few confinements with respect to transient perspectives in addition to parallel and interleaved circumstances where exercises depended on each other.

The model proposed by Liu and colleagues [29] mines temporal sequences as highlights to encode temporal relatedness among activities. It utilizes a versatile multi-errand learning strategy to catch the relationship among activities and to select the discriminant features. Nevertheless, these supervised methods for model construction needed to annotate the training data set.

Liu and colleagues [27] used the idea of lossless recovery image and used them in vision-based child-adult interaction behaviour with missing images due to infrastructural error. We applied this idea in our framework to guess the missing sensor noise with slight modification as highlighted in [26, 27].

A probabilistic system [28] for human movement is proposed by intertwining the low-level and high-level state approaches for conquering their constraints. The two methodologies continue in parallel and each time the framework combines their commitments by a probabilistic strategy which could supplement the benefits of both. Keeping in view the end goal to choose the trackers in light of the movement types, the calculation for trackers inspecting depends on low and high-dimensional trackers. The two techniques have been utilized individually for low-dimensional and high-dimensional trackers; they incorporated adaptively by sampling the trackers based upon the motion type.

**Table 1** Summarizing previous sensor-based activity recognition

Author	Modelling Techniques	Window size	Sequential Activities Recognition	Parallel Activities Recognition	Personalize/complete Activities Recognition
Chen et al. [9]	Knowledge Driven(ontology based)	Fixed	Yes	No	No
Okeyo et al. [38]	Knowledge Engineering+ temporal	Fixed	Yes	Yes	No
Ye et al. [49]	Data Driven(Pyramid Kernel) + ontology	Dynamic	Yes	Yes	No
Azkune et al. [3]	Hybrid(Data Driven+ knowledge Driven)	Dynamic	Yes	No	Yes
Riboni et al. [44]	Probabilistic model+ ontology based	Dynamic	Yes	Yes	No
Our approach	Hybrid(Knowledge Driven+ Data Driven+ temporal)	Dynamic	Yes	Yes	Yes

For achieving better performance, a multi-view learning system has been proposed [31] to influence the information and to combine the varied properties from different views to characterize objects and feature extractors. As there are various spatial and temporal data models around water quality stations, combining two different views with a station may achieve better performance.

Liu and colleagues [32] used a transparent fusion of data from multiple social networks to predict the futuristic career path. In activity recognition system the same situation regarding activities prediction may occur when there are missing sensors in the action stream.

Table 1 shows a comparison summary of promising knowledge-driven and hybrid AR techniques for parallel and interleaved activities.

### 3 Architecture of OSCAR

In order to provide a detailed insight into architectural components (as illustrated in Fig. 6), some important definitions have been devised for providing the rationale to discussion and further details.

**Definition 1: Sensor Activation:** Sensor activation is triggered when a sensor changes its state from *non-responsive* to *responsive*. For example, when an inhabitant takes a “cup”, the generated signal in the data stream is labelled as “cupSens”. The sensor stimulation consists of few properties such as timestamp ( $tS^\circ$ ), sensor id (Sen-id) and location ( $L^\circ$ ).

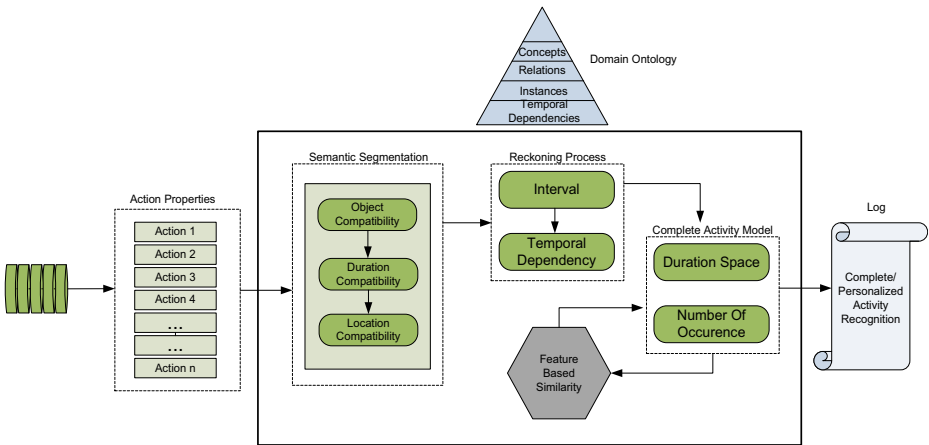
$$SS^\circ = \{\text{timestamp} + \text{sensor-id} + \text{location}\}$$

$$SS^\circ = \{tS^\circ + \text{Sen-id} + L^\circ\}$$

**Definition 2: Action Property:** Sensor activation is directly linked with action properties that make our AR model generic. For example, “cupSens” and “mugSens” are associated with action property “*hasDrinkingContainer*”. This action property is a part of the “making tea” activity. So “cupSens” and “mugSens” can be used alternatively during the activity of “making tea”.

$$\text{hasDrinkingContainer} = \{\text{cupSens}, \text{mugSens}\}$$





**Fig. 6** Detailed architecture of OSCAR

**Definition 3: Perceptible Activity Model (PAM):** It is a set of necessary action properties identified by a domain expert while performing certain activity. For example, the “making tea” activity consists of necessary action properties such as “hasDrinkingContainer”, “hasHeating”, “hasAdding” deduced from sensor activations such as “stoveSens”, “teaSens”, “milkSens” and “waterSens”, whereas “sugarSens” is an optional ingredient. So, the assumption in PAM is:

$$PAM_{Tea} = \{hasHeating(stoveSens), hasDrinkingContainer(cupSens), hasAdd(milkSens), hasAdding(teaSens)\}$$

**Definition 4: Optional Actions:** They are not mandatory in performing an activity but can be part of that activity for depicting a personalized behaviour of inhabitants. Optional actions play an integral role in recognition of Complete Activity Models.

**Definition 5: Complete Activity Model (CAM)/Personalized Model:** It is the set of all necessary and optional actions performed by certain inhabitant while performing an activity. The CAM may vary along each inhabitant.

$$CAM_{Tea} = \{turnOnTap(waterSens), hasAdding(milkSens), hasAdding(teaSens), hasDrinkingContainer(cupSens), hasHeating(stoveSens), hasAdding(sugarSens)\}$$

**Definition 6: Overlapped Actions:** They are the set of actions in a sensor stream corresponding to two different PAMs while performing different activities in same time frame.

$$Activity = \{S-action_{i1}, S-action_{i2} \dots S-action_{in}, S-duration_i\}$$

$$Activity = \{S-action_{j1}, S-action_{j2} \dots S-action_{jn}, S-duration_j\}$$

$$Overlapped\ Activity = \{S-action_{i1}, S-action_{i2}, S-action_{j1}, S-action_{j2}, S-action_{j2} \dots S-action_{in}, S-action_{jn}\}$$

**Definition 7: Missing sensor noise:** It is the inability of sensors to stimulate the data signals caused by infrastructural issues or sensor error despite of inhabitants’ genuine sensor interaction.

**Definition 8: Sensor Noise/Noisy Sensor/ Erratic Behaviour:** It refers to the sensor activation due to a user’s mistaken interaction with objects that are not part of any ongoing activity.

An abstract view of the operational components of OSCAR is illustrated in Fig. 6 with an elaboration at a finer level of granularity discussed in the following sub-sections.

Every interaction of an inhabitant with different objects, while performing activities over a temporal scale, is transmitted as a data stream from sensor to AR system for recognizing the current activities based upon contextual information in a domain ontology. Such domain ontology describes the hierarchy of the activities, objects and action properties by establishing the relationship among activities and objects (details in section 3.1). Once the data stream is received, the first step is to transform the sensor stream into action properties (details in section 3.2). Later, the semantic segmentation process identifies the generic activity model in the stream based on the context of the object, duration and location (elaborations in section 3.3). This process may yield a number of overlapped generic activity models. Some of these models are rightly identified as parallel activities while others may be wrongly asserted as generic models that need to be discarded. A reckoning process discards the spurious generic activity models by identifying action intervals and inferring their temporal dependencies (as explained in section 3.4). The next step is to identify the complete activity model by calculating the duration space dynamically (discussed in section 3.5). Complete activity models are further refined through feature-based similarity (given in section 3.6). Finally, the *complete activity models* that have been identified are modelled into a log file for behaviour analysis of the inhabitants.

### 3.1 Activity modelling in a domain ontology

Keeping the above definitions in view, ontology artefacts have been devised in the form of classes for an ontological activity model, as shown in Fig. 7.

*ADL Activities:* is super concept of all the simple activities performed at home like “making pasta”, “making tea”, “taking a bath”, etc. Each ADL activity is composed of a set of action properties and descriptive properties, such as the “making pasta” activity that requires the actions: “use stove”, “add water”, “add pasta” etc. Descriptive properties are used to describe the activity, such as the duration of the activity, the location to perform the activity etc.

*Object:* is a super concept of home commodities, electrical appliances, fixtures or other entities used in routine for performing different activities. Examples of these objects can be “stove”, “cup”, “fridge”, or “waterTap”.

*Sensors:* there are two types of sensors used in our model: On/Off-sensors such as “on/off stove” or “on/off waterTap”; and Contact-sensors that produce a time interval between a

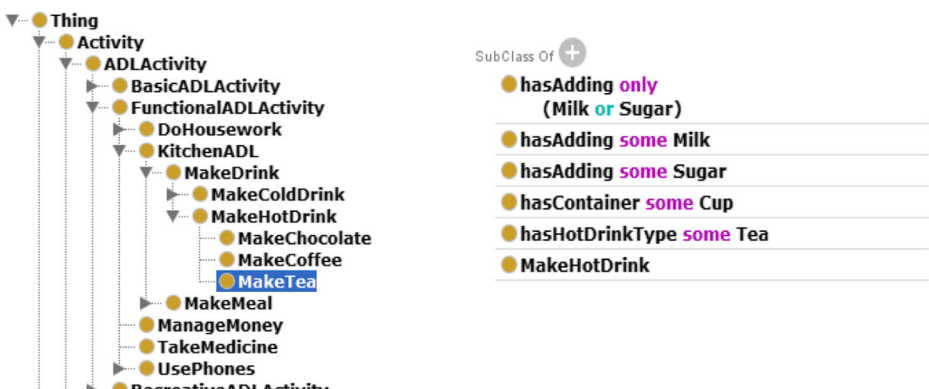


Fig. 7 Domain ontology for activity modeling (Make Tea)

start and end time recorded after certain object is touched. Each instance of a sensor class corresponds to an instance of an object class (or its subclasses) through the “attachedWith” property.

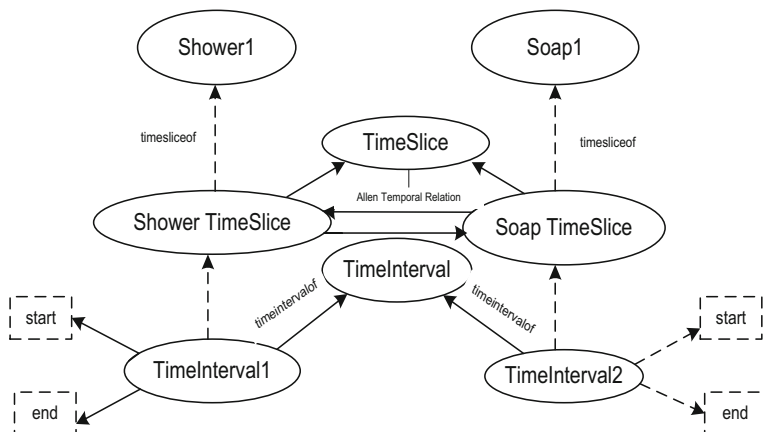
*Location:* is super concept of different locations in home, such as kitchen, bathroom, living room, etc. Each activity is bind to only one location where it can be performed and each object is bind to only one location where it can be used.

A snapshot of the developed ontology is illustrated in Fig. 7, which shows the action and description properties of the “making tea” activity.

### 3.1.1 Temporal concepts in activity modelling

The temporal relations among AR actions are required to identify a complete activity model. Usually the actions of an activity can be performed in any random order, but some of the actions in activities may have temporal dependencies on each other. For example, in the “bathing” activity, the action “use shower” may be used before or during the “use soap” action. Such temporal relations require a tertiary relation among instances of the concepts whereas ontologies typically hold a binary relation among the concepts. In order to accommodate these temporal relations [20, 36], a time ontology [22] and the 4D extended fluent approach [48] have been used. In smart homes, activities are partially static and diachronic in nature. Static parts of activities such as location of activities and objects used in activities can be represented in OWL. On the other hand, diachronic parts like the interval of the actions or the inter-dependency among the actions can be represented with time and the 4D extended fluent approach, as shown in Fig. 8. It illustrates a temporal relation between the “use shower” and “use soap” actions of the “bathing” activity.

Usually, actions of an activity are an open interval in nature. Open interval refers to the time interval where starting and ending points are not explicitly specified. In the process of recognizing the complete activity model, starting and ending points of activities are not known. The 4D extended fluent approach has proved effective in determining the qualitative relations among open interval actions in AR processes. Our approach has been implemented by using Allen’s temporal relations (pairwise disjoint). The main classes of temporal relations are:



**Fig. 8** Ontological dynamic activity model (Bathing)

*DateTimeDescripton*: this concept represents the temporal stamp of data activities and object interactions such as year, day hour, minute, or second.

*Duration*: describes the interval of an activity, such as 10 min for an activity to complete.

*TimeInterval*: is the domain of the time intervals class and *TimeSlice*: is the domain for entities representing temporal parts. “Time interval” instances are used to maintain the start and end time information of a slice. The property “timeSliceOf” builds the relation among the instances of the “TimeSlice” class and the “Object” class, while the “timeIntervalOf” property holds the relationship among the instances of two time intervals.

### 3.2 Mapping sensor activations to action properties

A stream of real sensor activations can be transformed into action properties [3] using the proposed domain ontology. Such transformation offers the advantages of Description Logic (DL) and OWL inference through relations of subsumption, equivalence and transitivity. Concepts related to “activities” in the ontology have super/sub relations with each other. Super and intermediate concepts are coarse-grained (abstract) activities while the leaf level concepts are fine-grained (concrete) activities. Fine-grained concepts inherit all the action properties and description properties from their super concepts in addition to their own properties and restrictions on inherited properties.

The conversion of sensor streams into action properties makes it more generalized in recognizing the patterns through inference instead of strict object patterns. An example of transforming the sensor stream into an action property is given next:

*Sensor stream*: {*cupSens*, *whiteSugarSens*, *skimmedMilkSens*}

Transformed stream: {*hasDrinkingContainer* (*cupSens*), *hasAdding*(*whiteSsugarSens*), *hasMilk* (*skimmedmilkSens*)}

The result of the sensor stream transformed above remains the same even if “*mugsSens*”, “*brownSugarSens*”, “*liquidMilkSens*” is used instead of “*cupSens*”, “*whiteSugarSens*” and “*skimmedMilkSens*”.

### 3.3 Semantic segmentation

Semantic segmentation is an iterative process that clusters the sensor stream by observing the actions associated with generic activity models (or perceptible activity models - PAM) modelled in ontology. The segmentation process considers the following aspects for completing an activity: if an action is part of multiple segments, then it is considered in every segment. If a segment has completed its set of actions, the segment will be marked as closed along with its duration. If an action is not part of any PAM, that action is labelled as “optional”. Few of the identified segments may be invalid (due to different issues to be discussed in the coming sections) and require a validation decision for retention or discarding. Validation of these segments is evaluated through a Semantic Segmentation(SS) process which is based on three aspects: (i) Object compatibility ( $O_c$ ), (ii) Duration compatibility ( $D_c$ ) and (iii) Location compatibility ( $L_c$ ).

$$SS_{(0,1)} = O_{c(0,1)} \wedge D_{c(0,1)} \wedge L_{c(0,1)} \quad (1)$$

**Object compatibility** Objects placed at home may be used in more than one activity. Our ontological model has an object property named “activity type”. It describes the possible activities in which a particular object may be used termed as object compatibility. For example, the object “stove” is used in different kitchen activities like “making tea”, “making coffee”, “making pasta”, etc.

**Duration compatibility** Each activity has an estimated time duration for completion; for example, the “making tea” activity has a duration of 5 min while “making pasta” has 10 min. Time duration starts from the first occurrence of an action of the PAM and ends with the time duration specified in the ontology. It is mandatory for all the actions of a PAM to complete within its specified duration.

**Location compatibility** Each activity is stamped with a location for example “making tea” is performed in the kitchen while “brushing teeth” is performed in the bathroom. During the course of our work, we proceed with the baseline assumption that the location of activities and their associated objects will always be the same. For example, objects used in “making tea”, such as “stove”, “sugar jar”, or “milk jar”, will always be located in the kitchen.

The algorithm for the semantic segmentation process is given below.

---

**Listing 1:** Semantic Segmentation Algorithm

---

```

Input =sensor stream S = {S1, S2, S3, S4, S5, S6...SN}, KD=Domain Knowledge Ontology.
Output = Pi = List of PAM's discovered during semantic segmentation Process.
Let ai are the list of actions in each PAM (Pi) = {a1, a2, a3 ....an}
begin
For each (Si)
Check →Si ∈ Pi's actions
IF      [Pi] → Open then
      add(Si) → [Pi]
Else If
      Open → New [Pi] → Label it with PAM's label described in ontology.
Else
      Label Si → “OPTIONAL”
End if
End Loop
For each (Pi)
  duration (Pi) = getLastActionTime() – getFirstActionTime()
  IF (duration (Pi) <= KD-Duration (PAM)) then
    For each ai ∈ Pi
      IF (Location (ai) == Location (Pi))
        Valid PAM
      End if
    End Loop
  End if
End Loop
Close [Pi]
Discard all Open [Pi]
Return (List of closed [Pi])
End

```

---

The complexity of the algorithm is  $O(N^2)$ . Here, it is worth mentioning that two overlapped activities may have the same location and common objects to perform activities with overlapped duration. This ambiguous situation may entail the lack of an AR decision by the semantic segmentation algorithm. The non-decisiveness of the semantic segmentation algorithm, with many overlapped PAMs may be caused due to the following reasons:

- (1) An inhabitant really performs parallel activities where the sensor stream comprises the mixed set of actions for multiple activities.
- (2) An action may be part of multiple activities for PAMs running in parallel. Partial sets of actions from different parallel activities may combine to produce new PAMs.
- (3) Overlapped segments may arise due to the sensor noisy/erratic behaviour. At some moment of time, an inhabitant interacts with some object mistakenly resulting in making clusters overlap. For example, the first coffee jar is contacted unintentionally and then the tea jar is interacted for the activity of “making tea”. As a result, the two PAMs “making coffee” and “making tea” are overlapped.
- (4) An action may be an indispensable part of one activity and may also be an optional part of another activity. For example, “spoon” is a mandatory part of the activity “making coffee”, while it is an optional part of the activity “making tea”.

In the above-mentioned situations, the most challenging task is to remove the anomalous/false/spurious clusters out of the identified segments. The reckoning process removes the anomalous/false/spurious segments to a larger extent though it does not manage to completely remove such erroneous segments. The reckoning process is used to further refine the process of removing anomalous clusters as discussed next.

### 3.4 Reckoning process

In the reckoning process, there are three temporal aspects to be considered when removing the anomalous segments.

Contact sensors have been used to monitor the interval of interaction with a particular object, in contrast to traditional binary sensors, producing certain values when an object is interacted.

The Time ontology [22] has been used to model the minimum interval for each action in the PAM of an activity; for example, in the “making tea” activity, “stove” has a minimum interval of 3 min while the “sugar jar” interval is 10 s. These minimum interval considerations for each corresponding action have three advantages:

- (i) Any interaction with a duration lower than the minimum interval refers to noisy sensor/erratic behaviour. A valid interval for certain action in an activity can be identified with the sum of intervals of more than one interaction with the same object. For example, if the stove is turned “OFF” after 1 min and then turned “ON” for two minutes again, a total time of 3 min is a valid action time for the “making tea” activity.
- (ii) If the action time of an activity exceeds its minimum interval and the same action is part of overlapping/consecutive activities, that action is considered as an isolated part of the two activities. For example, for consecutive/overlapped activities of “making tea” and

**Table 2** Allen temporal relations among the actions of an activity

Interval	Inverse Interval
$before(\Delta t_a, \Delta t_b) \rightarrow T_{af} < T_{aj}$	$after(\Delta t_a, \Delta t_b) \rightarrow T_{ai} > T_{bf}$
$meet(\Delta t_a, \Delta t_b) \rightarrow T_{af} = T_{bi}$	$meethy(\Delta t_a, \Delta t_b) \rightarrow T_{bf} = T_{ai}$
$overlap(\Delta t_a, \Delta t_b) \rightarrow T_{ai} < T_{bi} < T_{af}$	$overlapby(\Delta t_a, \Delta t_b) \rightarrow T_{bi} < T_{ai} < T_{bf}$
$Start(\Delta t_a, \Delta t_b) \rightarrow (T_{ai} = T_{bi}) \quad (T_{af} < T_{bf})$	$Startby(\Delta t_a, \Delta t_b) \rightarrow (T_{bi} = T_{ai}) \quad (T_{bf} < T_{af})$
$during(\Delta t_a, \Delta t_b) \rightarrow (T_{ai} > T_{bi}) \quad ((T_{af} < T_{bf}))$	$contain(\Delta t_a, \Delta t_b) \rightarrow (T_{ai} < T_{bi}) \quad ((T_{bf} < T_{af}))$
$finish(\Delta t_a, \Delta t_b) \rightarrow (T_{af} = T_{bf}) \quad (T_{ai} > T_{bi})$	$finishby(\Delta t_a, \Delta t_b) \rightarrow (T_{af} = T_{bf}) \quad (T_{bi} > T_{ai})$
$equal(\Delta t_a, \Delta t_b) \rightarrow (T_{ai} = T_{bi}) \quad (T_{af} = T_{bf})$	

“making coffee”, the stove “turned on” action with a duration of 10 min is treated as two segregated actions.

- (iii) In an object-sensor-based AR approach [9], the order of the actions does not matter; e.g., the actions of the “making tea” activity can be performed in any order. But it is observed in some activities that order needs to be maintained in few actions out of all the action set. For example, in the “taking bath” activity, “soap” must be used before or during the “showering” action. Similarly, in the “making coffee” activity, the “useSoap” action must be carried out after interacting with “kettle”, “coffee” and “mug”. Such type of open interval actions can be represented by Allen’s temporal relationships [1, 2].

Allen’s temporal relations have been used to model temporal dependencies among actions of activities where the order of actions is important. Allen’s relations have been used with the time ontology and 4D extended fluent to express the interval of the actions and their relations while performing different activities. The binary relations of Allen’s intervals used by OSCAR are given in Table 2:

Let:  $\Delta t$  is the duration space of an activity.

$\Delta t_a$  &  $\Delta t_b$  is the interval of actions “a” and “b”, respectively.

$T_{ai}$  &  $T_{bi}$  is the initial time of actions “a” and “b”, respectively.

$T_{af}$  &  $T_{bf}$  is the end time of actions “a” and “b”, respectively.

Here, it is worth mentioning that the presented binary relations can be extended further through inference using transitive relations. Once these relations are defined, the activities not complying with the order defined in the context knowledge of the domain ontology are discarded. This further refines the PAMs by removing spurious activities. Now, the refined sensor stream is analysed for recognizing a complete activity model specific to the inhabitants’ behaviour in the AR process.

### 3.5 Learning the complete activity model

In this step of the AR process, a complete activity model is identified by incorporating the actions labelled as “optional” during the semantic segmentation process as explained in section 3.3. This requires the actual duration space of the activity. The *duration space* of an activity refers to the temporal duration of the sensor stream under observation to gather the set of actions for completing a particular activity. Determining the duration space of an activity is a challenging task since the exact start and end times of an activity are not known for the Complete Activity Model (CAM). On the other hand, the start and end times of the Generic model (i.e., PAM) are easily identified in the semantic segmentation process. However, the

start and end times of the PAM cannot exactly be the start and end times of the Complete Activity Model (CAM) since optional actions have no order to be performed. Such random order of actions may occur before a PAM, during a PAM or after a PAM has been completed. Therefore, the duration space of an activity may be captured entirely such that all possible occurrences of actions for the current activity are known.

The duration space is modelled through the “duration” property in the OSCAR’s ontology model for holding enough intervals while completing an activity. Generally, the first action of complete activity precedes PAM’s first action and the last action follows PAM’s last action. Therefore, it is assumed that the first action of PAM is the first action of CAM where the end time of CAM is the first action of PAM plus the duration encoded in the ontology for certain activity. Similarly, assuming that the last action of PAM is the last action of CAM, the initial time of CAM is the last action of PAM minus the duration encoded in the ontology. This is how duration is dynamically computed in the surroundings of PAM. The duration space of an activity has been calculated through the following formula:

$$\Delta t = T_f - T_i \tag{2}$$

Where.

$$T_i = dt_f - \Delta_D.$$

$$T_f = dt_i + \Delta_D.$$

$\Delta t$  = Stipulated duration space of the Complete Activity Model (CAM).

$T_i$  = Initial time instance of the duration space of CAM.

$T_f$  = Final time instance of the duration space of CAM.

$\Delta_{dt}$  = Duration of the PAM in the sensor stream.

$dt_f$  = Final time instance of the PAM (first occurrence of the action in PAM).

$dt_i$  = Initial time instance of the PAM (last occurrence of the action in PAM).

$\Delta_D$  = Duration of PAM encoded in the ontology.

Now, all the actions labelled as “optional” within a duration space of respective activities are assigned to their semantically compatible PAMs. If an optional action does not exist in any duration space, it is termed as an outlier. An *outlier* must be the part of the following or preceding activity. For this, we propose a heuristic that calculates the centroid of both activities. The outlier is identified as part of the activity located nearest to the centroid, conditioning the action is semantically compatible. The centroid (C) of an activity is calculated by using the following formula:

$$C_{(Ai)} = T_i + (T_f - T_i) / 2 \text{ where } C \text{ is centroid of an Activity } A_i \tag{3}$$

$\Delta t$  can be calculated by the following algorithm.

---

**Listing 2:** Algorithm for calculating duration space of Complete Activity Model(CAM) with outlier

---

Input =Centroid of an activity  $C_{(Ai)}$  , occurring time of outlier (ot)

Output =  $\Delta t$  = Stipulated duration space of the Complete Activity Model (CAM)

Let  $o_i$  be the outlier action that occurs at time  $o_i$ .

if  $((C_{(Ai)} - o_t) < (C_{(Aj)} - o_t))$

    fuse( $o_i, A_i$ )

$\Delta t = O_t - T_i$

        Where  $T_i$  is the initial time of activity  $A_i$

else

    fuse( $o_j, A_j$ )

$\Delta t = O_t - T_f$

        Where  $T_f$  is the final time of activity  $A_j$

---



After the reckoning process completes, activities still have a chance of having spurious segments requiring further refinement by removing the noisy actions. For example sensor stream “SS” fulfils every criterion for two activities:

$$SS = \{teaSens, stoveSens, waterSens, coffeeSens, spoonSens, milkSens, sugarSens, strainerSens, creamerSens, cupSens, iceSens\}$$

The order of actions in the above sensor stream is temporally correct and semantically compatible with both activities of “making coffee” and “making tea”. But the time interval indicated by the actions infers that only one activity, either “making tea” or “making coffee”, is being performed or the second one exists only due to sensor noise. The  $\Delta t$  of an activity provides a way to distinguish such similar activity patterns or spurious patterns.

Here Optional Sensors (OS) are considered for performing an activity. Optional sensors may vote to an actual occurred activity. Such refinement has been performed through a feature-based similarity technique as elaborated in following section.

### 3.6 Feature-based similarity

Feature-based similarity addresses the issue of erratic overlapped PAMs based on optional sensors similarity. We have customized Tversky’s similarity [34] concepts for our activity recognition process as given below:

$$similarity(O1, O2) = \frac{\alpha(\psi(O1) \cap \psi(O2))}{\beta(\psi(O1) \setminus \psi(O2)) + \gamma(\psi(O2) \setminus \psi(O1)) + \alpha(\psi(O1) \cap \psi(O2))}$$

Where,

$\psi(O)$  is the function capacity depicting all the important highlights of the object O.

$\alpha, \beta$  &  $\gamma \in \mathbb{R}$  are real number constants. For  $\alpha = 1$  common features of the two objects have maximal importance and  $\beta = \gamma$ .

The following notations have been used in our work:

- Common features of PAM and OS:  $cf(PAM, OS) = \psi(PAM) \cap \psi(OS)$
- Distinctive features of PAM:  $df(PAM) = \psi(PAM) \setminus \psi(OS)$
- Distinctive features of OS:  $df(OS) = \psi(OS) \setminus \psi(PAM)$

Using the notation and setting  $\alpha = \beta = \gamma = 1$ , the above formula becomes:

$$SimT(PAM, OS) = \frac{cf(PAM, OS)}{(df(PAM) + df(OS) + cf(PAM, OS))} \tag{4}$$

To find the Common features for PAM and OS, the formula would be:

$$cf = \frac{k^2}{h_1 h_2} \quad df_1 = \frac{h_1 - k}{h_1} \quad \text{and} \quad df_2 = \frac{h_2 - k}{h_2}$$

Where  $K$  is the number of common features having PAM and OS whereas  $h_1$  is the different members from PAM and  $h_2$  is the different members from OS.

We assume that PAM and OS have common features based on an “activity type” property from a sensor stream (SS) as follows:

$$\begin{aligned} PAM_{Tea} &= \{tea, stove, water, milk, spoon\}. \\ PAM_{Coffee} &= \{coffee, stove, water, milk, spoon\}. \\ \text{and } OS &= \{sugar, strainer, creamer, cup, ice\}. \end{aligned}$$

In the above example, the values for tea are:  $K = 1$ ,  $h_1 = 4$  and  $h_2 = 3$ .

Distinctive sensors having the same value of the “activity type” property as of  $PAM = df_1(PAM)$ . Distinctive sensors having the same value of the “activity type” property as of  $Optional\ Sensor = df_2(OS)$ .

$$cf = 1/(5*3) = 0.067, df_1 = (5-1)/5 = 0.8 \text{ and } df_2 = (3-1)/3 = 0.67$$

Putting these values in eq. 4 for finding similarity between tea and its optional sensors:

$$Sim_{Tea} = 0.067/(0.8 + 0.67 + 0.067) = 0.044$$

The same formula can be used for “making coffee” after we get values  $K = 2$ ,  $h_1 = 5$  and  $h_2 = 2$ .

Common features for coffee and optional sensors are:

$$cf = 2/(5*2) = 0.2, df_1 = (5-2)/5 = 0.6 \text{ and } df_2 = (2-2)/2 = 0$$

Putting these values in eq. 2 for finding the similarity between tea and its optional sensors:

$$Sim_{Coffee} = 0.2/(0.6 + 0 + 0.2) = 0.25$$

The similarity value of “making tea” is 0.044 and the value of “making coffee” is 0.25. It implies that the activity sequence of “making tea” should be discarded asserting “making coffee” as the actual activity performed.

## 4 Dataset and evaluation methodology

In order to evaluate the performance of OSCAR and have an empirical view of its effectiveness compared with contemporary approaches; baseline datasets played an important role. The experiments were run on two separate datasets, namely CASAS by Cook et al. [45] and other acquired by our methodology named Data Acquisition Methodology for Smart Homes (DAMSH).

### 4.1 The CASAS dataset

There are very few datasets available comprehending the interleaved activities annotated for smart homes. The ‘Interleaved ADL Activities’ (IAA) dataset was selected from the CASAS smart home project [45] after a thorough analysis of activities. This data set was collected in a

smart apartment test bed hosted at Washington State University during the academic year 2009–2010.

The CASAS dataset covers interleaved ADLs of twenty-one inhabitants acquired in a smart home laboratory. Almost 70 sensors were used to collect data about movement, temperature, use of water, interaction with objects, doors and phones. Eight activities were considered: filling medication dispenser, watching DVD, watering plants, answering the phone, preparing birthday card, preparing soup, cleaning, and choosing outfit. The average time taken by the participants to complete the eight activities were 3.5 min, 7 min, 1.5 min, 2 min, 4 min, 5.5 min, 4 min, and 1.5 min, respectively. The average number of sensor events collected for each activity was 31, 59, 71, 31, 56, 96, 118, and 34, respectively [10].

The order and expenditure of time were up to the inhabitants and they were allowed to perform the activities in parallel. Only one person was allowed to stay in the home while acquiring the data. Five out of eight activities have been considered for evaluation. The five activities shortlisted are: *watching DVD*, *answering the phone*, *preparing birthday card*, *preparing soup* and *cleaning*. Three activities which have not been included in experiments are: *choosing outfit*, *watering plants* and *filling medication dispenser*. The reason for not considering these three activities is having few distinct actions in their action sequence. So it is not appropriate to build generic model and to learn for devising a complete model in our case. For example *choosing outfit activity* has just two distinct sensors in its dataset i.e. motion sensor and cabinet sensor. These actions are not sufficient to build the generic model and hence complete model.

We pre-processed the CASAS dataset to build the Perceptible Activity Model for all activities. All the distinct actions that had been used in dataset for a particular activity were enlisted. Necessary actions are chosen as a part of the PAM and the rest of the actions are considered as user-specific actions. For example, the “preparing soup” activity includes seven distinct sensor values: cabinet sensor, water sensor, burner sensor, raisin sensor, oatmeal sensor, pot sensor, and motion sensor. Out of seven sensors, four were considered part of the PAM such as cabinet sensor, water sensor, burner sensor and pot sensor while the rest of sensors (such as raisin sensor, oatmeal sensor) were considered as optional ones.

## 4.2 The DAMSH dataset

The rationale for acquiring the Data Acquisition Methodology for Smart Homes (DAMSH) is given in the following.

Firstly, datasets used for deriving a complete/personalized activity model from a generic activity model are not available publicly on dataset sources like the CASAS Datasets [27], Box lab [28] or any other AR portals [39] to the best of our knowledge.

Secondly, the proposed solution (OSCAR) claims to recognize highly similar activities. Such similar activities have one or two different actions in the generic model. For example, “making tea” and “making coffee” can be differentiated with only one action, i.e., “add tea” and “add coffee”; the rest of the actions in both activities are the same. A single positive sensor noise may cause a serious challenge in recognizing the activities so considering such scenarios in AR is very important. Such scenarios are not available in any of the datasets to the best of our knowledge.

Thirdly, the available datasets do not have positive sensor noise caused by unintentional interactions. One of the research challenges addressed by OSCAR is to identify and remove

the positive sensor noise from the sensor stream. DAMSH comprehends all these scenarios while assuring the coverage of common scenarios in AR at smart homes.

#### 4.2.1 Data acquisition in DAMSH

One of the famous methodologies [42, 46] is used by DAMSH with different steps as follows: (i) Select a home to install sensors over home objects; (ii) Select the inhabitants residing to perform the activities; (iii) Label the segmentations of sensor streams generated as result of inhabitants' actions; (iv) Use the labelled data as ground truth for AR, for evaluation process and to build the activity ontology (as described in section 3.1); (v) Use the same datasets to test the AR system and to store the labels produced; and (vi) Compare the labels of the AR system with ground truth using appropriate metrics.

The idea starts with the process of getting the real time user input through an activity survey form targeted to the inhabitants. The survey form recorded the possible steps for performing the same activities differently along with the duration to perform the activities. Different aspects of the activities were also considered such as the most probable day interval of performing an activity, e.g., taking a bath between 7:00 AM to 8:00 AM; the location of the activity, e.g., kitchen, bathroom, etc. Moreover, activities that inhabitants perform in parallel are enlisted such as “making tea” and “toasting” while taking breakfast. The purpose of circulating activity surveys is to encode the context-based knowledge of the activities in the ontology such as the necessary actions, duration and temporal dependencies. Moreover, survey form data has been used for generating the ground truth data through simulation. The survey form is available online:

<https://www.dropbox.com/s/yddl2om2xz5w0j9/Human%20activity%20recognition%20Survey.docx?dl=0>

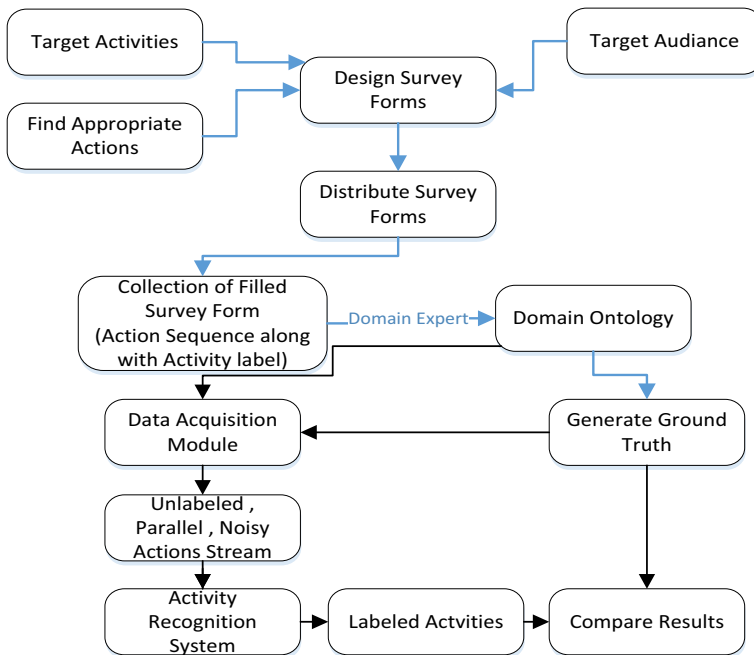
Fig. 9 illustrates the data flow diagram of the data acquisition module for DAMSH.

Targeting the right audience and their activities plays a key role in designing the survey forms. The target groups for the survey were selected from communities having specialization in performing different activities, for example chefs, house hold ladies performing kitchen activities, or launderer for washing and ironing activities. The survey forms with responses had labelled activities and action sequences. Domain experts developed the domain ontology based upon perceptions of performing activity sequences as discussed in section 3.1. It was hard to incorporate all the action variants of an activity in the survey form so a simulation tool was developed that received the activity scripts and domain ontology as input and produced the permutation of action sequences of all activities (called ground truth). Ground truth facts, unlabelled data acquisition source files and dataset file are available online:

<https://www.dropbox.com/sh/28e7ca7gdeap7d9/AAAFGhlcXmdEmJIYh5fTxAVya?dl=0>

The purpose of the proposed AR models is to recognize a complete action sequence for parallel activities. So, a handful of parallel, unlabelled action sequences along with some noise (see definition 8) was desired. Table 3 shows the characteristics and statistics of DAMSH with associated details.

DAMSH acquired 2.483 sensors stimulations in 111 days. A total of 10 distinct activities have been performed in parallel and sequential fashion. Among the parallel activities are “watching tv”, “making tea”, “making coffee”, and “making pasta”. Among the sequential activities are “taking nap”, “chores”, “shaving”, “bathing”, “taking medicine”, and “washing cloth”. The dataset contains 10% sensor noise of actual data. Sensor noise is unbiased and



**Fig. 9** Data flow diagram for data acquisition and evaluation

generated without any human interruption by the random() function. DAMSH code, sensor noise stimulation code and software manual are available online:

<https://www.dropbox.com/sh/vis7da2hi0f8fa9/AABSoRDpUuqpCKIDOEXy4hiEa?dl=0>.

The output of this module is labelled datasets having: (i) variation of ground truth by adding sensor noise; and (ii) Parallel activities by mixing the action stream of two or more activities.

## 5 Results and discussion

The performance of OSCAR is measured through accuracy metrics [47] such as True Positive (TP), False Positive (FP) and False Negative (FN). Definitions of TP, FP, and FN in the context of Activity recognition as given in the following:

*True Positive:* The sequence action of an activity is labelled correctly by the system.

*False Positive:* An activity did not occur but it is labelled incorrectly by the system.

*False Negative:* The sequence action of an activity exists but is not identified by the system.

Other experiments are done with F-measure [47], an accuracy metric that uses two parameters: precision and recall. Precision is the number of the times that an activity is correctly identified divided by the number of times the activity is inferred while Recall is the number of the times that an activity is correctly inferred divided by number of times that it really happens. As an overall measure of accuracy, F-measurement has been used that is a balanced measure of precision and recall together. Also, OSCAR has been compared with

**Table 3** Data set statistics of DAMSH

Data set characteristics	No. of days.	Total No. of sensor occurrence
<ul style="list-style-type: none"> <li>• Randomize sequence of actions for activities of each day</li> <li>• Contains 10% sensor noise of actual data</li> <li>• Unbiased data, generated by using contextual knowledge of activity ontology, and sensor noise by random() function</li> <li>• Single variant</li> </ul>	111	2,483
<p>Parallel activities</p> <ul style="list-style-type: none"> <li>• Watching tv with making tea</li> <li>• Watching tv with making pasta</li> <li>• Making tea with making coffee</li> <li>• Making pasta with making tea</li> </ul>	<p>No. of activities occurrence</p> <ul style="list-style-type: none"> <li>• making pasta(70)</li> <li>• making tea(111)</li> <li>• making coffee(37)</li> <li>• watching Tv(111)</li> <li>• taking nap(134)</li> <li>• chores(100)</li> <li>• shaving(70)</li> <li>• bathing(50)</li> <li>• taking medicine(30)</li> <li>• washing cloth(10)</li> </ul>	<p>Sensor noise</p> <p>10% of actual data set</p>

prevalent techniques in the context of F-measurement to see its impact over the CASAS and DAMSH datasets (on noisy as well as noiseless data).

The impact of the different modular units of OSCAR such as Semantic Segmentation (SS), Reckoning Process (RP) and Feature-based Similarity (FS) has also been covered in the evaluation. These units of OSCAR are compared with respect to their accuracies on a noisy/noiseless dataset. Results based on these metrics and scenarios are tabulated below followed by the necessary discussion.

The performance of Semantic Segmentation (SS) module is assessed with noisy and noiseless data as given in Table 4.

Table 4 depicts the result of the semantic segmentation process on noisy and noiseless data for sequential activities. In sequential activities, all the actions occur within a specific time frame considered to be the part of an ongoing activity or to be a noisy action. Since noiseless data streams are well refined (free of noise) and with an unambiguous set of actions, Semantic Segmentation presents an ideal performance on noiseless data in terms of identifying correct clusters of the perceptible activity model. Table 4 (column-1) shows 100% true positivity on noiseless stream. But the accuracy seems a compromise when there is noise (definition 8) in a stream and PAM actions are highly similar with other PAMs. As a result, true positive and false positive get affected. For example, the accuracy of “making tea” and “making coffee” are compromised in TP and FP for noisy data. The “making tea” activity is performed 130 times while “making coffee” is performed 60 times in 111 days. The results observed in the TP ratio of noisy data are 90 and 92%, respectively, which means that the AR system is capable of segmenting the activity for 117 times and 55 times, respectively, while having noise in the data stream. The presence of FP in both the “making tea” and “making coffee” activities with 10 and 8% is due to a higher degree of similarity among the action sequence of two activities. A single unintentional touch with the “making coffee” object during the “making tea” activity resulted in generating the “making coffee” segment and vice versa. Other three activities having lower degree of TP are “watching TV”, “taking medicine” and “chores”. This lower degree of TP is due to a small number of actions in their PAMs where a single positive sensor noise or missing sensor may produce FP and FN, respectively. The activities “making pasta”, “bathing” and “washing cloth” have given consistent results on noisy and noiseless data because the noise is semantically recognized in these activities with the factor of location and the used-in property.

**Table 4** Semantic segmentation process for sequential activities on noisy and noiseless data

Activities	True Positive %		False Positive%		False Negative%	
	Noiseless	Noise	Noiseless	Noise	Noiseless	Noise
Making Tea	100	90	0	10	0	0
Making Coffee	100	92	0	8	0	0
Making Pasta	100	100	0	0	0	0
Bathing	100	100	0	0	0	0
Shaving	100	100	0	0	0	0
Watching TV	100	80	0	10	0	10
Taking Medicine	100	85	0	10	0	5
Washing Cloth	100	100	0	0	0	0
Taking Nap	100	95	0	5	0	0
Chores	100	90	0	10	0	0

The second experiment assesses the performance of the Reckoning Process (RP) and the Feature-based Similarity (FS) process in OSCAR with noisy and noiseless data for the parallel activity recognition.

In Table 5, the performance of the Perceptible Activity Model (PAM) recognition is presented with an insight and impact of modules in OSCAR such as the Reckoning Process (RP), which eliminates the spurious PAM by exploiting the temporal dependency among the actions of an activity, if any. Feature-based Similarity (FS) further augments the elimination process by measuring the similarity of a segment with user-specific actions in that cluster.

The performance of RP degrades if two activities are highly similar in their action sequence, occurring at same place in parallel fashion and their temporal dependency criteria are fulfilled. For example, the scenarios of “making tea” and “making coffee”, as discussed in section 3.6. As the RP module fails to recognize the spurious PAM, the FP factor increases. Similarly, the “making pasta” activity has a TP of 70%. Most of the actions of “making pasta” are also common with “making tea” and “making coffee” where use-spoon, add water, use-stove are used commonly. A few mistaken object interactions may generate spurious activity entailing in a higher FP. Similarly, “taking bath”, “shaving” and “washing cloth” have many actions in common in their PAMs. Results are promising for those activities whose actions are distinct from other activities or when activities are performed at different locations. The TP of the activity model shows significant improvement when the output of the RP module is processed with the Feature-based Similarity process. Feature-based similarity is processed on the basis of user-specific actions voting for a particular PAM in a specific duration. This module refines the RP module results. As we see, the TP for the “making tea” activity improves from 65 to 93% and “making coffee” improves from 67 to 92%. It is worth mentioning that FN values appear in the results due to missing sensor or infrastructural error and is out of the scope of this paper.

The third experiment assesses the performance of leaning the Complete Activity Model (CAM) with noisy and noiseless data for parallel activity recognition.

In Table 6, the performance of OSCAR after identifying complete activity models (CAM) is presented over a mix of noisy and noise-free datasets. The promising performance shown by CAM is due to the dynamic duration (window) space instead of the static duration encoded in the ontology. The dynamic window caters the actions of an activity that are outlier of any activity duration. FP and FN appear in some cases due to actions that lie outside the boundaries of the dynamic duration or due to missing sensor noise (described in definition 7).

**Table 5** Perceptible model recognition comparison on noisy and parallel activities using Reckoning Process (RP) and feature based similarity (FS)

Activities	True Positive%		False Positive%		False Negative%	
	RP	FS	RP	FS	RP	FS
Making Tea	65	93	25	3	10	3
Making Coffee	67	92	27	5	6	3
Making Pasta	70	91	29	3	1	6
Bathing	77	95	33	0	0	5
Shaving	73	97	26	0	1	3
Watching TV	86	99	14	1	0	0
Taking Medicine	89	97	11	1	0	2
Washing Cloth	90	99	10	0	0	1
Taking Nap	98	100	2	0	0	0
Chores	96	100	4	0	0	0

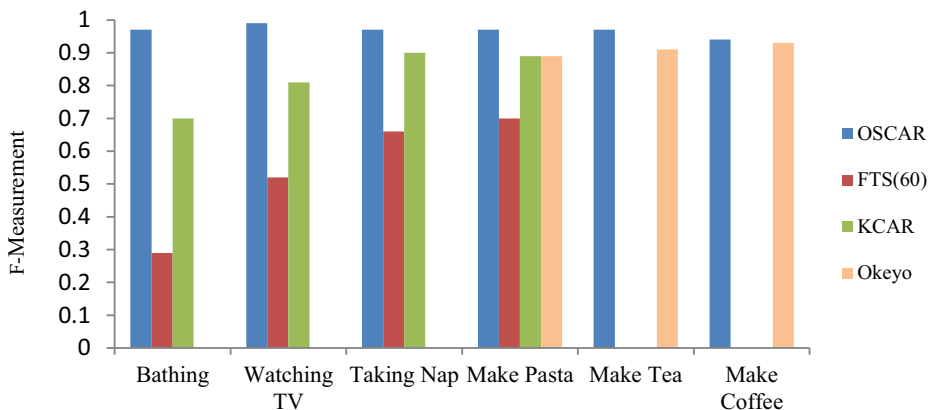


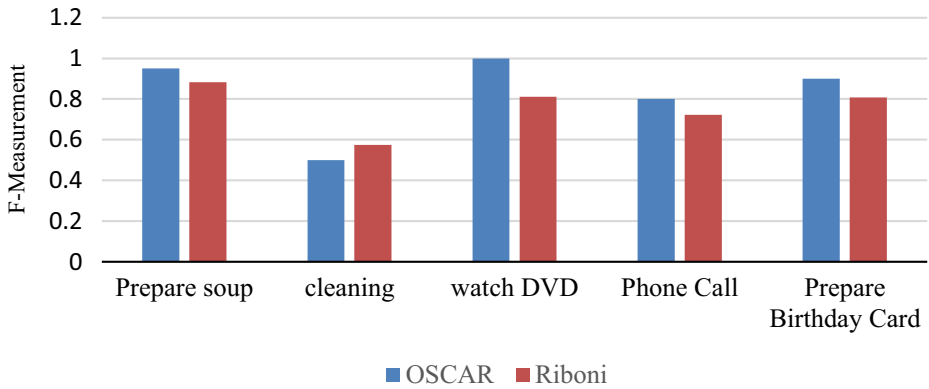
**Table 6** Complete recognition model for noisy and noiseless parallel scenario

Activities	True Positive		False Positive		False Negative		Accuracy
	Noise-less	Noise	Noise-less	Noise	Noise-less	Noise	
Making Tea	96	95	1	1	3	4	0.95
Making Coffee	95	94	0	1	5	5	0.94
Making Pasta	98	97	1	1	1	2	0.97
Bathing	97	97	0	0	3	3	0.97
Shaving	98	98	0	0	2	2	0.98
Watching TV	99	99	1	1	0	0	0.99
Taking Medicine	99	99	1	1	1	1	0.99
Washing Cloth	99	99	0	0	1	1	0.99
Taking Nap	100	97	0	0	0	3	0.97
Chores	100	98	0	0	0	2	0.98

To further demonstrate the effectiveness of the proposed approach, we compared the complete activity recognition results of OSCAR with well-known techniques described in the literature [38, 49], as shown in Fig. 10. F-measure has been used as performance metric. Fig. 10 shows the F-measure result on the DAMSH dataset while Fig. 11 shows the F-measure result on the CASAS dataset. Fig. 10 compares the activities of “making tea”, “making coffee” and “making pasta” with [38] while “making pasta”, “bathing”, “watching TV”, “taking nap” is compared with [49]. The results in Fig. 10 show that OSCAR outperforms the rest of the techniques described in [38, 49]. Specifically, OSCAR has better performance than FTS due to the dynamic duration approach increasing AR accuracy compared to FTS that uses a static sliding window approach. This fact can be observed in Fig. 10 where FTS exhibits poor performance for the “bathing” activity since this activity takes 2 min as defined in the ontology duration property compared to the actual time taken by the inhabitant, i.e., more than 2 min. Similarly, FTS shows better performance for the “making pasta” activity since the time defined in the ontology is of 5 min compared with the actual time of 4 min.

KCAR recognized the boundary of parallel activities more accurately using distance measured through Least Common Subsume (LCS) for having a smaller number of partitions that better resemble real activities. The fundamental standard is to segregate sensor occurrence and gathering similar ones, yet it does not ensure that the assembled sensor events map to only

**Fig. 10** Comparison of accuracy with other knowledge driven techniques



**Fig. 11** Comparison of accuracy with Riboni [44]

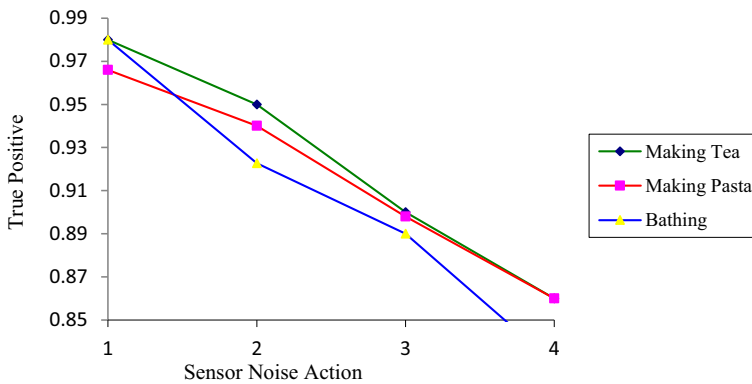
one activity. KCAR’s performance degrades when the activity actions cannot be distinguished like bathing, which is not well separable from shaving and washing hands, or making pasta, which is not well separable form making tea and making coffee. Okeyo and colleagues’ results are promising and near to the proposed work; they differ slightly on all activities including “making tea”, “making coffee” and “making pasta”. The OSCAR methodology focused on personalized activity modelling while Okeyo and colleagues worked on composite activity modelling.

Fig. 11 shows the results on the CASAS dataset as described in section 4.1. Results are compared with a state of the art technique [44]. To process the CASAS dataset by our system, a domain ontology model was developed for the activities given in the dataset: watch DVD, answer the phone, prepare birthday card, prepare soup, and cleaning. All distinct actions were enlisted from the dataset for each activity. All the necessary and user-specific actions of each activity were identified to build the perceptible activity model for each activity and encode all the actions in the domain ontology. The results in Fig. 11 show that OSCAR performed better than the technique described [44] for experimented except the cleaning activity. Since PAM actions of the cleaning activity are a super set of all other activities, the cleaning activity results in interaction with almost all the objects such as cabinet, oatmeal, raisin, pot, burner, medicine, bowl, brown sugar, etc. All these objects are used in the rest of the activities and end up with spurious actions showing a degraded performance for the cleaning activity.

The performance of OSCAR needs to be measured with positive sensor noise with noisy data stream. In principle, the semantic segmentation criterion of the activity suggests that if the action sequence of an activity is mapped to one of the PAM, the semantic segmentation process marks it as a performed activity. Mistaken interactions with wrong objects may result

**Table 7** Positive sensor noise effect on accuracy

Activities	No. of Sensors	Positive Sensor Noise Effect			
		1 TP FP FN	2 TP FP FN	3 TP FP FN	4 TP FP FN
Making Tea		.98 .02 0	.95 .05 0	.90 .11 0	.86 .14 0
Making Pasta		.98 .02 0	.94 .06 0	.89 .11 0	.86 .14 0
Bathing		.96 .04 0	.92 .08 0	.89 .11 0	.83 .17 0



**Fig. 12** The TP of activities in presence of positive sensor noise

in a number of overlapped spurious PAMs over the original one that affect the TP rate. It implies that the greater the positive sensor noise, the lower the TP rate.

Table 7 shows the results for scenario having positive sensor noise in the dataset. Three activities were shortlisted for this experiment: “making tea”, “making coffee” and “bathing”. These experiments were run while having the noisy data of 111 days. The results tabulated in Table 7 have been illustrated in Fig. 12 having “Sensor Noise Action” on the x-axis and “Accuracy (True Positivity)” on the y-axis. Activity-based evaluation criterion has been used for having TP, FP and FN rates. However, the TP rate of the “making tea”, “making coffee” and “bathing” activities seems to reduce linearly with positive sensor noise. TP reduction causes an increase in the FP rate as the detected activities did not occur actually. There is no effect on FN but the effect on TP is apparent as expected.

## 6 Conclusion

This paper presents a novel approach named OSCAR for deriving a complete activity model for parallel activities from a generic activity model. The effectiveness of the proposed approach is signified by the semantics in perspective of duration, location, activity type, temporal dependency among activity actions and feature-based similarity among activity actions. The complete activity model characteristics claimed in the introduction section have been achieved by different components of the proposed model. (i) During the evaluation, it has been observed that sensor noise produced subtle anomalies during parallel activities recognition which are removed gradually by the semantic segmentation, reckoning process and feature based similarity; (ii) The proposed algorithm in semantic segmentation is not influenced by the order of the occurrence actions in a sequence. The order among the actions is considered during the reckoning process only if it is explicitly defined in the ontology through the 4D fluent approach. The rest of the actions are catered in all possible randomized orders. (iii) Our proposed approach does not use a conventional static time window approach; instead, a dynamic calculation of the AR duration is proposed that played a key role in identifying the complete/personalize activity model. (iv) Similarly, if an object is part of multiple activities, the feature-based similarity component calculates the action similarity for an activity if two activities are running in parallel.

OSCAR is presented for single-user parallel activities with capacity for extending and recognizing multiple user activities in collaborative manner. Also, these activities can be extended from simple to composite ones. We look forward to complete the AR process for inhabitants to extend their behaviour for learning and evolving the context knowledge (modelled in ontology) by identifying the specialized activities performed by inhabitants. Lastly, the set of activities in CASAS dataset which have not been considered in experiments due to infrastructural issues will be incorporated in future experiments with necessary arrangements.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

## References

1. Allen JF (1983) Maintaining knowledge about temporal intervals. *Commun ACM* 26.11:832–843
2. Allen JF, Ferguson G (1994) Actions and events in interval temporal logic. *J Log Comput* 4.5:531–579
3. Azkune G et al (2015) Extending knowledge-driven activity models through data-driven learning techniques. *Expert Syst Appl* 42.6:3115–3128
4. Bao L, Intille SS (2004) Activity recognition from user-annotated acceleration data. In *International Conference on Pervasive Computing* (pp. 1–17). Springer Berlin Heidelberg
5. Caragliu A, Del Bo C, Nijkamp P (2011) Smart cities in Europe. *J Urban Technol* 18(2):65–82
6. Chen L, Nugent C (2009) Ontology-based activity recognition in intelligent pervasive environments. *Int J Web Info Syst* 5(4):410–430
7. Chen L, Nugent CD, Wang H (2012) A knowledge-driven approach to activity recognition in smart homes. *IEEE Trans Knowl Data Eng* 24(6):961–974
8. Chen L, Hoey J, Nugent CD, Cook DJ, Yu Z (2012) Sensor-based activity recognition. *IEEE Trans Syst Man Cybern Part C Appl Rev* 42(6):790–808
9. Chen L, Nugent C, Okeyo G (2014) An ontology-based hybrid approach to activity modeling for smart homes. *IEEE Trans Human-Mach Syst* 44(1):92–105
10. Cook D, Schmitter-Edgecombe M, Crandall A, Sanders C, Thomas B (2009) Collecting and disseminating smart home sensor data in the CASAS project. *Proc CHI Workshop Dev Shared Home Behav Datasets Adv HCI Ubiquitous Comput Res*: 1–7
11. Cui J, Liu Y, Xu Y, Zhao H, Zha H (2013) Tracking generic human motion via fusion of low-and high-dimensional approaches. *IEEE Trans Syst, Man, Cybernet: Syst* 43(4):996–1002
12. Díaz-Rodríguez N, Cadahia OL, Cuéllar MP, Lilius J, Calvo-Flores MD (2014) Handling real-world context awareness, uncertainty and vagueness in real-time human activity tracking and recognition with a fuzzy ontology-based hybrid method. *Sensors* 14(10):18131–18171
13. Gayathri KS, Elias S, Shivashankar S (2015) Composite activity recognition in smart homes using Markov logic network. *IEEE Computer Society*
14. Gayathri KS, Elias S, Shivashankar S (2015) Composite activity recognition in smart homes using Markov logic network. *Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom)*, 2015 IEEE 12th Intl Conf on, pp. 46–53. IEEE
15. Georis, B., Maziere, M., Bremond, F., and Thonnat, M., (2004) A video interpretation platform applied to bank agency monitoring. *Proc 2<sup>nd</sup> workshop Intell Distrib Surv Syst*: 46–50
16. Gu T, Wu Z, Tao X, Pung HK, Lu J 2009, March. epsicar: an emerging patterns-based approach to sequential, interleaved and concurrent activity recognition. *Pervasive Comput Commun*, 2009. *PerCom 2009. IEEE Int Conf* (pp. 1–9). IEEE
17. Gu T, Wang L, Wu Z, Tao X, Lu J (2011) A pattern mining approach to sensor-based human activity recognition. *IEEE Trans Knowl Data Eng* 23(9):1359–1372
18. Gubbi J, Buyya R, Marusic S, Palaniswami M (2013) Internet of things (IoT): a vision, architectural elements, and future directions. *Future Gen Comput Syst* 29(7):1645–1660
19. Hakeem A, Shah M (2004) Ontology and taxonomy collaborated framework for meeting classification. *Proc Int Conf Pattern Recogn*: 219–222
20. Helaoui R, Niepert M, Stuckenschmidt H (2011) Recognizing interleaved and concurrent activities using qualitative and quantitative temporal relationships. *Perva Mobile Comput* 7.6:660–670

21. Helaoui R, Niepert M, Stuckenschmidt H (2011) Recognizing interleaved and concurrent activities: A statistical-relational approach. *Perva Comput Commun (PerCom)*, 2011 IEEE Int Conf (pp. 1–9). IEEE
22. Hemalatha M, Uma V, Aghila G (2012) Time ontology with reference event based temporal relations (retr). *Int J Web Semantic Technol* 3:1
23. <http://boxlab.wikispaces.com/List+of+Home+Datasets> (accessed April 2018)
24. Kakde A, Gulhane V (2015) Real time composite user activity modelling using hybrid approach for recognition. In *Electrical, Computer and Communication Technologies (ICECCT)*, 2015 IEEE International Conference on (pp. 1–6). IEEE
25. Khan S, Safyan M (2014) Semantic matching in hierarchical ontologies. *J King Saud Univ-Comput Inf Sci* 26(3):247–257
26. Likavec S, Cena F (2015) Property-based semantic similarity: what counts?. AIC
27. Liu Y, Zhang X, Cui J, Wu C, Aghajan H, Zha H (2010) Visual analysis of child-adult interactive behaviors in video sequences. *16th Int Conf Virtual Syst Multimed (VSMM) 2010*:26–33 IEEE
28. Liu Y, Cui J, Zhao H, Zha H (2012) Fusion of low-and high-dimensional approaches by trackers sampling for generic human motion tracking. *Pattern Recogn (ICPR)*, 2012 21st Int Conf (pp. 898–901). IEEE
29. Liu Y, Nie L, Han L, Zhang L, Rosenblum DS (2015) Action2Activity: recognizing complex activities from sensor data. *IJCAI*: 1617–1623
30. Liu L, Cheng L, Liu Y, Jia Y, Rosenblum DS (2016) Recognizing complex activities by a probabilistic interval-based model. *AAAI* 30:1266–1272
31. Liu Y, Zheng Y, Liang Y, Liu S, Rosenblum DS (2016) Urban water quality prediction based on multi-task multi-view learning
32. Liu Y, Zhang L, Nie L, Yan Y, Rosenblum DS (2016) Fortune teller: predicting your career path. *AAAI*: 201–207
33. Lu Y, Wei Y, Liu L, Zhong J, Sun L, Liu Y (2017) Towards unsupervised physical activity recognition using smartphone accelerometers. *Multimed Tools Appl* 76(8):10701–10719
34. McKeever S, Ye J, Coyle L, Bleakley C, Dobson S (2010) Activity recognition using temporal evidence theory. *J Ambient Intell Smart Environ* 2.3:253–269
35. Meditskos G, Kontopoulos E, Kompatsiaris I (2015) ReDef: context-aware recognition of interleaved activities using OWL 2 and defeasible reasoning. In *SSN-TC/OrdRing@ ISWC*: 31–42
36. Milea V, Frasinca F, Kaymak U (2008) Knowledge engineering in a temporal semantic web context. *Web Eng, 2008. ICWE'08. Eighth Int Conf. IEEE*
37. Nevatia R, Hobbs J, Bolles B (2004) An ontology for video event representation. *Proc IEEE Workshop Event Detect Recognit*: 119
38. Okeyo G, Chen L, Wang H, Sterritt R (2012) A knowledge-driven approach to composite activity recognition in smart environments ubiquitous computing and ambient intelligence vol 7656 of the series lecture notes in computer science. Springer, Berlin Heidelberg, pp 322–329
39. Online referencing <http://ailab.wsu.edu/casas/datasets/> (accessed April 2018)
40. Pantelopoulos A, Bourbakis NG (2010) A survey on wearable sensor-based systems for health monitoring and prognosis. *IEEE Trans Syst Man Cybern Part C Appl Rev* 40(1):1–12
41. Poppe R 2010. A survey on vision-based human action recognition. *Image and vision* Pantelopoulos, A. and Bourbakis, N.G., 2010
42. Riboni D, Bettini C (2011) COSAR: hybrid reasoning for context-aware activity recognition. *Pers Ubiquit Comput* 15(3):271–289
43. Riboni D, Pareschi L, Radaelli L, Bettini C (2011) Is ontology-based activity recognition really effective?" *Pervasive Computing and Communications Workshops (PERCOM Workshops)*, IEEE International Conference on. IEEE
44. Riboni D, Sztaylor T, Civitarese G, Stuckenschmidt H (2016) Unsupervised recognition of interleaved activities of daily living through ontological and probabilistic reasoning. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (pp. 1–12). ACM
45. Singla G, Cook DJ, Schmitter-Edgecombe M (2009) Tracking activities in complex settings using smart environment technologies. *Int J Biosci Psychiatr Technol (IJBSP)* 1(1):25
46. Tapia EM, Intille SS, Larson K (2004) Activity recognition in the home using simple and ubiquitous sensors. In *International conference on pervasive computing* (pp. 158–175). Springer, Berlin, Heidelberg
47. Van Kasteren T, Alemdar H, Ersoy C (2011) Effective performance metrics for evaluating activity recognition methods. In *ARCS 2011 VDE*
48. Welty C, Fikes R, Makarios S (2006) A reusable ontology for fluent in OWL. *FOIS* 150:226–236
49. Ye J, Stevenson G, Dobson S (2015) KCAR: a knowledge-driven approach for concurrent activity recognition. *Perva Mobile Comput* 19:47–70



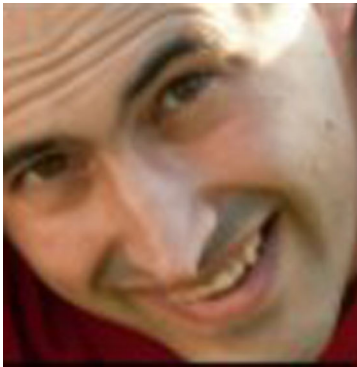
**Muhammad Safyan** Author did his MS in Information Technology from “National University of Science and Technology” (NUST), Islamabad, Pakistan and is currently doing his PhD from University of Gujrat, Gujrat, Pakistan. Author is working as Assistant Professor in Department of Computer Science, GC University, Lahore, Pakistan.



**Dr. Zia ul Qayyum** Author completed his doctoral degree in Artificial Intelligence from university of Leeds, UK. Research interests in the areas of Knowledge Representation, Data Mining, Recommendation Systems, and Semantic Information Retrieval. Currently working as Vice Chancellor, University of Gujarat, Pakistan.



**Sohail Sarwar** Author did his MS in Information Technology from “National University of Science and Technology” (NUST), Islamabad, Pakistan and is currently doing his PhD from University of Gujrat, Gujrat, Pakistan. Author is working as Assistant Professor in Department of Computer Science, GC University, Lahore, Pakistan.



**Raul Garcia Castro** Assistant Professor at Universidad Politécnica de Madrid, Spain. Author is post-doctoral researcher and is active member of ontology engineering group.



**Mehtab Ahmed** Author did his MS from GC University Lahore, and working as computer Programmer in GC University, Lahore