D5.3
Behavioural Profile Learning

Dementia Ambient Care: Multi-Sensing Monitoring for Intelligent Remote Management and Decision Support

Dem@Care - FP7-288199

eHealth
Better Healthcare for Europe
SEVENTH FRAMEWORK PROGRAMME
European Commission
Information Society and Media

Deliverable Information

Project Ref. No.		FP7-288199
Project Acronym		Dem@Care
Project Full Title		Dementia Ambient Care: Multi-Sensing Monitoring for Intelligence Remote Management and Decision Support
Dissemination level:		Public
Contractual date of delivery:		Month 23, 30 September 2013
Actual date of delivery:		15 October 2013
Deliverable No.		D5.3
Deliverable Title		Behavioural Profile Learning
Type:		Report
Approval Status:		Final
Version:		1.1
Number of pages:		67
WP:		WP5 Medical Ambient Intelligence
Task:		T5.2 Patient-tailored Dynamic Knowledge Enrichment
WP/Task responsible:		INRIA
Other contributors:		CERTH, UB1
Authors (Partner)		Carlos Crispim (INRIA), Serhan Cosar (INRIA), Francois Bremond (INRIA), Georgios Meditskos (CERTH), Stamatia Dasiopoulou (CERTH), Charalambos Doulaverakis (CERTH), Aurélie Bugeau (UB1), Vincent Buso (UB1), Jenny Benois-Pineau (UB1)
Responsible Author(s)	**Name**	Carlos Crispim (INRIA), Serhan Cosar (INRIA), Francois Bremond (INRIA)
	Email	francois.bremond@inria.fr
Internal Reviewer(s)		Ceyhun Burak Akgul (Vistek ISRA Vision)
EC Project Officer		Gerard Cultot
Abstract **(for dissemination)**		This document presents the current work carried out with respect to behavioural profile learning for the purpose of supporting the patient-customised services targeted within Dem@Care. The objective of behavioural profile learning is to dynamically discover person-specific behaviour patterns that can be utilised to improve the recognition of the performed activities and to allow for PwD-tailored behaviour interpretation and assessment. These behaviour patterns may encapsulate a variety of aspects, including the manner in which daily activities are performed, idiosyncratic and habitual knowledge, as well as recurrent routines. In this report, we present the first efforts towards the aforementioned directions. Two approaches for discovering, modelling and recognising ADL are proposed, a supervised one using egocentric video data as input, and an unsupervised one using as input data from a fixed camera. The goal is to allow for the discovery of behaviour patterns through machine learning. In addition, an ontology-based pattern-oriented approach is presented for capturing in a formal manner higher-level behavioural aspects that encapsulate richer semantics.

Version Log

Version	Date	Change	Author
0.1-0.3	30/07/2013	Document Structure Proposition	Carlos Crispim (INRIA), Serhan Cosar (INRIA), Georgios Meditskos (CERTH), Jenny Benois (UB1), Francois Bremond (INRIA)
0.4	30/08/2013	Section 2.2, Section 4, contribution to conclusions Section	Aurélie Bugeau, Vincent Buso, Jenny Benois-Pineau (UB1)
0.5	25/09/2013	Section 2.1, Section 3, contribution to conclusions Section	Georgios Meditskos, Stamatia Dasiopoulou Chalarambos Doulaverakis (CERTH)
0.6	27/09/2013	Abstract, Executive Summary, Introduction Section, Section 2.3, Section 5, contribution to conclusion and merge of individual contributions	Carlos Crispim, Serhan Cosar (INRIA)
0.7	27/09/2013	Revisions to Executive Summary, Introduction and Conclusion Sections	Stamatia Dasiopoulou, Georgios Meditskos (CERTH)
0.8	27/09/2013	Final draft for internal review	Carlos Crispim, Serhan Cosar (INRIA)
0.9	30/09/2013	Internal review feedback	Ceyhun Burak Akgul (Vistek ISRA Vision)
0.10	08/10/2013	Address internal review comments	Stamatia Dasiopoulou, Georgios Meditskos, Charalambos Doulaverakis (CERTH)
0.11	08/10/2013	Address internal review comments	Aurélie Bugeau, Vincent Buso, Jenny Benois-Pineau (UB1)
1.0	11/10/2013	Address review comments – Final version	Carlos Crispim, Serhan Cosar, Francois Bremond (INRIA), Stamatia Dasiopoulou, Georgios Meditskos (CERTH)
1.1	15/10/2013	Address PMB comments	Carlos Crispim, Serhan Cosar (INRIA)

Executive Summary

This document reports on the current work carried out with respect to behavioural profile learning for the purpose of supporting the patient-customised services targeted within Dem@Care.

The objective of behavioural profile learning is to dynamically discover person-specific behaviour patterns that can be utilised to improve the recognition of the performed activities and to allow for PwD-tailored behaviour interpretation and assessment. These behaviour patterns may encapsulate a variety of aspects, including the manner in which daily activities are performed, idiosyncratic and habitual knowledge, as well as recurrent routines.

This deliverable reports on the first efforts towards the aforementioned directions. More specifically, two approaches for discovering, modelling and recognising ADL are proposed, a supervised one using egocentric video data as input, and an unsupervised one using as input data from a fixed camera. The goal is to allow for the discovery of behaviour patterns through machine learning that would otherwise be difficult to capture in a declarative way. In addition, an ontology-based pattern-oriented approach is presented for capturing in a formal manner higher-level behavioural aspects that encapsulate richer semantics.

Preliminary results of the supervised and unsupervised activity recognition approaches are presented on public datasets and Dem@Care dataset according to data availability of each sensor. Future work will extend the current evaluation by using a larger set of participants from the Dem@Care pilots according to the progress of system implementation at the pilot sites, and the proposed approaches according to the challenges presented at the pilot environments. A preliminary investigation of the proposed ontology-based behaviour patterns as means for capturing background knowledge in the semantic behaviour interpretation framework described in D5.2 has also been carried out.

Abbreviations and Acronyms

ADL	Activities of Daily Living
DnS	Descriptions and Situations
DUL	DOLCE UltraLite[1]
MOG	Mixture of Gaussians
MPEG	Moving Picture Experts Group
OWL	Ontology Web Language
OWL-QL	Ontology Web Language Query Language
OWL-DL	Ontology Web Language Description Language
PwD	People with Dementia
RDF	Resource Definition Framework
RGBD	Red-Green-Blue-Depth
SWRL	Semantic Web Rule Language
W3C	World Wide Web Consortium
XML	eXtensible Markup Language

[1] http://www.loa.istc.cnr.it/ontologies/DUL.owl

Table of Contents

List of Figures

List of Tables

1 Introduction

In order to support personalised health status assessment and in turn patient-tailored adaptation of feedback services as targeted by Dem@Care, it is quintessential to provide mechanisms for patient profiling as well as for updating and customising the clinical guidelines, recommendations and management procedures pertinent to people with dementia. It is the role of Task 5.2, to cater for such aspects.

Patient profiling involves the identification and extraction of behavioural patterns so as to enable: i) the recognition of the monitored PwD activities, despite idiosyncratic particularities in the way each individual carries out a given activity, and ii) the assessment of whether the monitored behaviour comprises abnormal or alarming deviations in PwD. Behavioural patterns may refer to low-level traits (e.g. walking pattern) as well as to higher-level behavioural aspects including habits and routines (e.g. the manner in which an individual makes his/her morning tea).

This deliverable reports on the first efforts towards the aforementioned directions. More specifically, two approaches for discovering, modelling and recognizing ADL are proposed, a supervised one using egocentric video data as input, and an unsupervised one using as input data from a fixed camera. In addition, an ontology-based pattern-oriented approach is presented for capturing in a formal manner high-level behavioural aspects. The population of these models can be done either manually by explicitly defining the modelled attributes, e.g. habits, preferences, interests, how a person performs an activity/order of activities, or by inferring unobservable information (patterns/trends) from observable data relating to their activities; currently the focus is on the former.

The rest of the deliverable is structured as follows. In Section 2, we present the state of the art methods for identification and extraction of behavioural patterns as a resource to build behavioural profiles. Section 3 presents the supervised activity recognition approach for egocentric cameras. Section 4 presents the proposed unsupervised framework for discovering, modelling and recognizing ADL. In Section 5, we describe the ontology-based patterns that have been developed for behaviour modelling. Finally, Section 6 concludes the deliverable and discusses next steps.

2 Related work

We herein present knowledge-driven methodologies applicable to identification and extraction of behavioural patterns as a resource to build behavioural profiles.

2.1 Unsupervised Learning of Activity Models using fixed Cameras

Activity analysis and recognition from video is a fast-growing field based on different methods and techniques. The goal of activity recognition is to analyse human activities from an unknown video based on the actions and movements of a person. A complete overview of previous methods on human activity recognition is proposed in many survey papers [2.1.1, 2.1.5, 2.1.14]. In [2.1.1, 2.1.5, 2.1.14], the authors emphasize the importance of high-level activity understanding for several important applications, namely those related to Activities of Daily Livings (ADLs).

As explained in Section 2.2, activity recognition can be performed by using egocentric wearable cameras. In addition, there are methods that address the problem of detecting complex daily activity using fixed cameras [2.1.16, 2.1.8, 2.1.12]. Based on the strategy of action classification, these methods can be categorized in various ways.

Previous works in [2.1.7, 2.1.10] in activity recognition are categorized as knowledge and logic-based approaches. For example, authors in [2.1.16] proposed a monitoring system for the analysis and recognition of human activities. It includes detecting and tracking people and recognizes their posture. Then, activities of interest are recognized based on this information. Three sources of knowledge were exploited: model of activities, 3D model of the observed scene and 3D model of the mobile object present in the observed scene. While a logic-based method is a natural way of incorporating domain knowledge, it requires an extensive enumeration by a domain expert for every deployment.

Recently, particular attention has been drawn on the object trajectory information over time to understand long-term activities. Trajectory-based approach can be classified into supervised and unsupervised methods: The supervised methods [2.1.6, 2.1.9] can build activity models in an accurate way. But they require manually labelled large training datasets. The unsupervised methods include works such as in [2.1.8] where authors use fuzzy k-means algorithm to cluster trajectories using spatial and temporal information and obtain motion patterns that are represented as a chain of Gaussian distributions. Based on the learned motion patterns, using the Bayes rule, the probability of a new trajectory under each motion pattern is calculated and it is used to detect anomalies and predict behaviours. In [2.1.4], the trajectories are modelled as a sequence of directions computed (roughly) via the angle between two consecutive positions. The sequence of directions is modelled as a Von Mises (or circular normal) distribution and, in an unsupervised way, the k-medoids clustering technique is used to build a mixture of Von Mises distributions. As in [2.1.8], the probability of a new trajectory under this distribution is used to detect abnormal behaviours. These methods depend on the final distribution's ability to represent actions. Therefore, these methods cannot represent and recognize events that have complex hierarchical structure in space and time. The approach in [2.1.12] uses HMM to represent trajectory paths and, based on eigenvector-based clustering, captures spatio-temporal relations in trajectory paths, allowing high-level analysis of an activity, which is suitable for detecting abnormalities. Zhong et al. divide the video into equal length segments and classify the trajectory features into prototypes that are obtained by k-means clustering [2.1.15]. Then, a prototype–segment co-occurrence matrix is computed from these prototypes and used to detect unusual events. Since the main goal of this approach is to detect abnormalities, the long-term trajectories are considered and the activities are modelled in coarse level. However, recognizing various types of daily living activities requires to model actions in space and time from coarse level to finer levels.

In addition, there are some methods that model the co-occurrences of actions. In [2.1.11] from RGB-D data, the positions of 15 body joints are extracted and used to recognize 16 actions using k-means clustering. Spatial-temporal motion features are encoded by spatial-temporal correlograms capturing long-range temporal co-occurrence patterns in [2.1.13]. Then, an unsupervised generative model is applied in order to learn different classes of human actions from these correlograms. Other than that, Bobick and Wilson use dynamic programming based approaches to classify activities [2.1.2]. These methods are effective when time ordering constraints hold.

In our approach, we propose a new framework that enables to model, discover and recognize activities in an unsupervised manner for monitoring patients. We use an intermediate level representation of features (the Primitive Events) composed of basic action primitives, which form the human motion. We present a hierarchical activity model categorizing complex activities using increasing granularity levels of the spatio-temporal structure of basic actions.

References

[2.1.1] J. K. Aggarwal and M. S. Ryoo, “Human activity analysis,” ACM Computing Surveys, vol. 43, pp. 1–43, 2011.

[2.1.2] A. F. Bobick and A. D. Wilson, “A state-based approach to the representation and recognition of gesture,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, pp. 1325–1337, 1997.

[2.1.3] J. Y. Bouguet, “Pyramidal implementation of the affine Lucas Kanade feature tracker description of the algorithm,” Intel Corporation, 2001.

[2.1.4] S. Calderara, R. Cucchiara, and A. Prati, “Detection of abnormal behaviors using a mixture of Von Mises distributions,” in 2007 IEEE Conference on Advanced Video and Signal Based Surveillance, 2007, pp. 141–146.

[2.1.5] A. A. Chaaraoui, P. Climent-Pérez, and F. Flórez-Revuelta, “A review on vision techniques applied to Human Behaviour Analysis for Ambient-Assisted Living,” Expert Systems with Applications, vol. 39, no. 12, pp. 10873–10888, Sep. 2012.

[2.1.6] S. Gong and T. Xiang, “Recognition of group activities using dynamic probabilistic networks,” in Proceedings Ninth IEEE International Conference on Computer Vision, 2003, vol. 2, pp. 742–749 vol.2.

[2.1.7] S. Hongeng, R. Nevatia, and F. Bremond, “Video-based event recognition: activity representation and probabilistic recognition methods,” Computer Vision and Image Understanding, vol. 96, pp. 129–162, 2004.

[2.1.8] W. Hu, X. Xiao, Z. Fu, D. Xie, T. Tan, and S. Maybank, “A system for learning statistical motion patterns.,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, pp. 1450–1464, 2006.

[2.1.9] I. Laptev and T. Lindeberg, “Space-time interest points,” in Proceedings Ninth IEEE International Conference on Computer Vision, 2003, vol. pages, pp. 432–439 vol.1.

[2.1.10] G. Medioni, I. Cohen, F. Bremond, S. Hongeng, and R. Nevatia, “Event detection and analysis from video streams,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, pp. 873–889, 2001.

[2.1.11] W.-H.Ong, L. Palafox, and T. Koseki, “Investigation of Feature Extraction for Unsupervised Learning in Human Activity Detection,” Bulletin of Networking, Computing, , Systems, and Software, vol. 2, no. 1, pp. 30–35, 2013.

[2.1.12] F. Porikli, “Learning object trajectory patterns by spectral clustering,” in Multimedia and Expo, 2004. ICME’04. 2004 IEEE …, 2004, pp. 2–5.

[2.1.13] S. Savarese, A. DelPozo, and J. C. Niebles, “Spatial-Temporal correlations for unsupervised action classification,” 2008 IEEE Workshop on Motion and video Computing, pp. 1–8, Jan. 2008.

[2.1.14] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea, “Machine recognition of human activities: A survey,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 18, no. 11, pp. 1473–1488, 2008.

[2.1.15] H. Zhong, J. Shi, and M. Visontai, “Detecting unusual activity in video,” in Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004., 2004, vol. 2, pp. 819–826.

[2.1.16] N. Zouba, F. Bremond, and M. Thonnat, “An Activity Monitoring System for Real Elderly at Home: Validation Study,” in Advanced Video and Signal Based Surveillance AVSS 2010 Seventh IEEE International Conference on, 2010, vol. 0, pp. 278–285.

2.2 Supervised Learning of Activity Models using Wearable Cameras

2.2.1 Wearable Cameras

Activity recognition has been widely studied by the computer vision community. We refer the reader to the surveys [2.2.1, 2.2.2] for a more complete treatment of the topic. Most of existing works are dedicated to videos captured by fixed ambient cameras. In the context of ADL recognition, using fixed cameras present some limitations. The field of view can be limited and might not always allow capturing all relevant information. It is therefore difficult to keep all relevant body parts, including hands, in focus and at sufficient resolution at all times.

Egocentric videos captured by wearable cameras present the advantage to capture the action realized by a person by always being focused on the user point of view. Hence, occlusions of manipulated objects tend to be minimized as the workspace containing the objects is always visible to the camera [2.2.3]. Furthermore, the same objects tend to be at the same viewing position when manipulated (generally the centre of the picture).

Wearable cameras also represent a cheap and effective way to record user activity for scenarios such as tele-medicine or life-logging.

Several devices have been proposed in the literature. As an example, the SenseCam device [2.2.4, 2.2.5], worn by a person, provides image lifelogs. The WearCam project [2.2.6] uses a camera strapped on the head of young children. This setup allows capturing the field of view of the children together with their gaze in order to monitor the impairments in child development. In [2.2.7, 2.2.8] a vest was adapted to be the support of the camera. The camera is fixed on the shoulder of the patient with hook-and-loops fasteners which allow the camera's position to be adapted to the patient's morphology. A microphone is integrated in the camera case and records a single audio channel.

2.2.2 Recognition of ADLs

There has been a fair amount of work on recognizing ADL by analysing egocentric videos. Most of the studies were performed under a constrained scenario, in which all the subjects wearing the cameras perform actions in the same room such as kitchen or office [2.2.9, 2.2.10], and therefore interact with the same objects, e.g., a hospital scenario of Dem@Care in which the medical staff ask patients to perform several activities. This scenario allows making assumptions on the objects or even uses instance-level visual recognition [2.2.11]. Fathi and al. [2.2.12] present a model for learning objects and actions with very little supervision. In particular a representation for egocentric actions based on hand-object interactions is introduced. The authors also develop an approach for automatically constructing a joint model of activities, actions and objects, in which the context provided by each element helps recognizing the others. In [2.2.9], observation stemming from hand motion templates and external sensors for room transitions are fed to a dynamic Bayesian network that infers the activity from a set of predefined sequences of recognized manipulations. In [2.2.7], a video structuring approach was introduced combining automatic motion based segmentation of the video and activity recognition by a hierarchical HMM. Both audio and visual features were used.

It is only recently that the more challenging unconstrained scenario has been examined regarding activity recognition. Kitani and al. [2.2.13] recognize ego-actions in outdoor environments using a stacked Dirichlet Process Mixture model.

Pirsivash and Ramanan [2.2.14] propose to train classifiers for activities based on temporal pyramids. These pyramids extend the well-known spatial pyramid to approximate temporal correspondence. Given a set of frames T to be analyzed and Kobject models, a score for an object i at pixel location and scale $\mathrm{p} = (\mathrm{x}, \mathrm{y}, \mathrm{s})$ in frame t is given by $\mathrm{score}_{\mathrm{i}}^{\mathrm{t}}(\mathrm{p}) \in [0,1]$.

The authors use the deformable part model [2.2.15] to compute these scores on record the maximum value of each model i in each framet:

$$f_{\mathrm{i}}^{\mathrm{t}}(\mathrm{p}) = \max_{\mathrm{p}} \mathrm{score}_{\mathrm{i}}^{\mathrm{t}}(\mathrm{p}).$$

The windows locations and scale which give the best scores are considered to host an object and are considered as features for the activity recognition task.

Bag of features is a natural way of aggregating these features through time. However, these representations ignore any temporal structure, e.g. “making tea” requires first “boiling water” and then “pouring it into a cup”. Inspired by the important work of [2.2.16] on spatial pyramid match kernel, the authors introduce temporal pyramids to address the use of different objects over time. At top-level $j = 0$ the feature is a histogram over the full temporal extent of a video clip $T^{0,0}$. The next level is the concatenation of two histograms obtained by temporally segmenting the video into two half ($T^{1,0}$ and $T^{1,1}$), and so on, leading to the representation:

$$\forall\, k \in \{1 \ldots 2^{j}\} x_i^{j,k} = \frac{2^{j-1}}{|\mathrm{T}|} \sum_{t \in T^{j,k}} f_{\mathrm{i}}^{\mathrm{t}}.$$

This model is then used for activity recognition by learning linear SVM classifiers on features

$$x = min\left(\left[x_1^{0,0} \ldots x_i^{j,k} \ldots x_K^{L,2^L}\right]^T, 0.01\right).$$

In a second part of the paper [2.2.14], the authors propose active object models to recognize more easily objects undergoing hand manipulations. In the training phase a subset of active training images for a particular object is added. To model active objects, the position and scale are added to the previous local appearance feature. Indeed, active objects tend to be at a position and scale convenient for hand manipulation. Applying this knowledge on the object being interacted with dramatically increases the performance.

The approach making use of these "active" areas for ADL recognition has also been studied by Fathi and al. [2.2.3] under a constrained scenario, where the authors enhanced the performance of their algorithm by defining visual saliency maps. They focus on activities requiring eye-hand coordination and model the relationship between the gaze point g_t, the scene objects and the action label a.For each pixel of an image, three features are used, leading to the final feature x_t:

- Object-based features, i.e. maximum scores of different object classifiers
- Appearance features, i.e. histogram of colour and texture on a circle around the pixel
- Future manipulation features; In general, hands activity in a few frames ahead provides a strong cue for predicting the gaze location in the current frame. First each frame of the video is segmented into foreground/background [2.2.17]. To check if a pixel in current frame f belongs to the foreground in frame $f + t$, the foreground mask of frame $f + t$ is transferred to frame $f$using optical flow vectors between adjacent frames.

For each action a, an SVM classifier is trained by selecting the positive features from the pixels surrounding the gaze locations in training sequences corresponding to the action a. The negative features are taken from the pixels far from the gaze point. After the classifiers are trained, the likelihoods $p(x_t|\mathrm{a}, \mathrm{g_t})$ can be computed. The final step of the algorithm consists in evaluating the posterior probability $p(a|X)$ of action a given the sequence of image features $X = \{x_1, \ldots, x_N\}$. This probability directly results from the previously computed likelihoods and the estimation of the most likely sequence of gaze locations.

Interactions between humans and objects have also been studied for classical videos. In [2.2.18], the human is first automatically detected at each frame. Next, the relevant object and its interaction with the human are determined. Only still images annotated with the action label are used for learning. No information on location of humans or objects is given to the classifier. The major contribution of this paper versus previous work that already studied human-object interaction [2.2.19, 2.2.20] is the use of a weakly supervised learning phase. This method has later been extended by including temporal information [2.2.21]. The object and the person are localized in space and tracked through time. An action is then represented as the trajectory of the object with respect to the person position.

References

[2.2.1] D. A. Forsyth, O. Arikan, L. Ikemoto, J. O'Brien et D. Ramanan, «Computational studies of human motion: part 1, tracking and motion synthesis,» *Found. Trends. Comput. Graph. Vis.,* vol. 1, n°12-3, pp. 77-254, 2005.

[2.2.2] P. K. Turaga, R. Chellappa, V. S. Subrahmanian et O. Udrea, «Machine Recognition of Human Activities: A Survey.,» *IEEE Trans. Circuits Syst. Video Techn.,* vol. 18, n° %111, pp. 1473-1488, 2008.

[2.2.3] A. Fathi, Y. Li et J. M. Rehg, «Learning to recognize daily actions using gaze,» chez *Proceedings of the 12th European conference on Computer Vision - Volume Part I*, Berlin, Heidelberg, 2012.

[2.2.4] S. Hodges, L. Williams, E. Berry, S. Izadi, J. Srinivasan, A. Butler, G. Smyth, N. Kapur et K. R. Wood, «SenseCam: A Retrospective Memory Aid.,» chez *Ubicomp*, 2006.

[2.2.5] D. Byrne, A. R. Doherty, G. J. F. Jones, A. F. Smeaton, S. Kumpulainen et K. J\"{a}rvelin, «The SenseCam as a tool for task observation,» chez *Proceedings of the 22nd British HCI Group Annual Conference on People and Computers: Culture, Creativity, Interaction*, 2008.

[2.2.6] L. Piccardi, B. Noris, O. Barbey, A. Billard, G. Schiavone, F. Keller et C. von Hofsten, «WearCam: A head mounted wireless camera for monitoring gaze attention and for the diagnosis of developmental disorders in young children,» chez *IEEE International Symposium on Robot & Human Interactive Communication*, 2007.

[2.2.7] S. Karaman, J. Benois-Pineau, V. Dovgalecs, R. Megret, J. Pinquier, R. Andre-Obrecht, Y. Gaestel et J.-F. Dartigues, «Hierarchical Hidden Markov Model in Detecting Activities of Daily Living in Wearable Videos for Studies of Dementia,» *CoRR,* 2011.

[2.2.8] R. Mégret, D. Szolgay, J. Benois-Pineau, P. Joly, J. Pinquier, J.-F. Dartigues et C. Helmer, «Wearable video monitoring of people with age Dementia : Video indexing at the service of helthcare.,» chez *International Workshop on Content-Based Multimedia Indexing*, 2008.

[2.2.9] S. Sundaram et W. Cuevas, «High level activity recognition using low resolution wearable vision,» chez *CVPR Workshops 2009.*, 2009.

[2.2.10] A. Fathi, A. Farhadi et J. M. Rehg, «Understanding egocentric activities,» chez *International Conference in Computer Vision*, 2011.

[2.2.11] P. Sturm, S. Ilic, C. Cagniart, S. Hinterstoisser, N. Navab, P. Fua et V. Lepetit, «Gradient Response Maps for Real-Time Detection of Textureless Objects,» *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 34, n° %15, pp. 876-888, 2012.

[2.2.12] A. Fathi, X. Ren et J. M. Rehg, «Learning to recognize objects in egocentric activities,» chez *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011*, 2011.

[2.2.13] K. Kitani, T. Okabe, Y. Sato et A. Sugimoto, «Fast unsupervised ego-action learning for first-person sports videos,» chez *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[2.2.14] H. Pirsiavash et D. Ramanan, «Detecting Activities of Daily Living in First-person Camera Views,» chez *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[2.2.15] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester et D. Ramanan, «Object Detection with Discriminatively Trained Part-Based Models,» *IEEE Transactions on Pattern Analyisis and Machine Intelligence,* vol. 32, n° %19, pp. 1627-1645, 2010.

[2.2.16] S. Lazebnik, C. Schmid et J. Ponce, «Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,» chez *IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[2.2.17] X. Ren et C. Gu, «{Figure-Ground Segmentation Improves Handled Object Recognition in Egocentric Video},» chez *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[2.2.18] A. Prest, C. Schmid et V. Ferrari, «Weakly supervised learning of interactions between humans and objects,» *{IEEE Transactions on Pattern Analysis and Machine Intelligence},* vol. 34, n° %13, pp. 601-614, 2012.

[2.2.19] B. Yao et L. F. fei, «L.: Modeling mutual context of object and human pose in human-object interaction activities,» chez *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[2.2.20] A. Gupta, A. Kembhavi et L. S. Davis, «Observing Human-Object Interactions: Using Spatial and Functional Compatibility for Recognition,» *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 31, n° %110, pp. 1775-1789, 2009.

[2.2.21] A. Prest, V. Ferrari et C. Schmid, «{Explicit modeling of human-object interactions in realistic videos},» *{IEEE Transactions on Pattern Analysis and Machine Intelligence},* vol. 35, n° %14, pp. 835-848, 2012.

2.3 Ontology-based Approaches for Behaviour Modelling

The inherent requirements in pervasive environments for heterogeneous data communication and integration have motivated a growing body of research in ontology-based frameworks for context modelling [2.3.1], key elements of which are *behavioural* aspects, such as daily activities, recurrent routines (e.g. bed time routine), frequency of activities, and so forth. The core idea is to map low-level information (e.g., objects, postures, location, and atomic events) and behaviour aspects on ontology models that capture everyday common sense knowledge about the individual's preferences and styles for performing certain activities. These models can then be used by ontology reasoning services to infer complex situations, detect abnormalities, learn new habits/trends, etc. In this section, we review the literature on ontology-based behaviour models, i.e., the data structures that can hold the characteristic attributes of a type of users. The population of these models can be done either manually by explicitly defining these attributes, e.g. habits, preferences, interests, how a person performs an activity/order of activities, or by inferring unobservable information (patterns/trends) from observable data relating to their activities. The definition of the data structures/ontology schema is usually referred to as *behavioural profile modelling* that serves as a template for manually defining or automatically inferring/recognising specific user profiles for different individual behaviours [2.3.2].

Several ontology-based models have been proposed in the literature to capture user behaviour in terms of *activity patterns* that describe the structure of complex activities that are built from atomic and other complex activities. Central to the proposed approaches, yet primary cause of variation is the approach taken to define such patterns. Roughly speaking, two strands permeate the relevant literature.

On one hand, there are ontology-based approaches that adhere to a purely ontology-based paradigm (e.g. Web Ontology Language 2 (OWL 2) [2.3.3]) using Description Logics [2.3.4] to model the semantics of the domain. Due to the lack of temporal semantics and limited support for non tree-like relations [2.3.5], such approaches cannot capture the temporal extension of activities nor intricate activity patterns. In particular, complex activity definitions are reduced to the atemporal intersection of their constituent parts [2.3.6][2.3.7][2.3.8][2.3.9] or at best, augmented, with notions such as *recently used* and *second last activity* to simulate some basic temporal reasoning [2.3.10]. As a result, purely ontology-based approaches fall short to effectively capture:

- Composite activities and information about their temporal extension [2.3.11], since OWL does not allow the assertion of new named individuals.
- Temporal (e.g. sequential, interleaved, concurrent) and non tree-like contextual correlations among activities, since the schema-level axioms in OWL cannot describe arbitrary relational structures, e.g., relations among individuals that are not connected.

On the other hand, hybrid approaches embrace the combination of ontologies and rules, as a way to compensate for the limitations of OWL and to provide for more expressive and flexible solutions than their purely ontology-based counterparts. Under this paradigm [2.3.12][2.3.13], ontologies are used to model the domain activities as a hierarchy of classes, with each class described by a number of properties, such as time and location, whereas rules are used to establish the complex activity correlations and the relative temporal extensions that define the activity pattern semantics. Prominent examples include the use of SWRL

(Semantic Web Rule Language) [2.3.14][2.3.15] and Jena [2.3.16] rules, the combination of ontologies with Complex Event Processing engines [2.3.17] and RDF (Resource Description Framework) stream reasoning [2.3.18][2.3.19][2.3.20]. For example, the activity recognition procedure in [2.3.16] is realised by a set of rule-based activity patterns in the Jena framework that combine ontological information relevant to location, sensors, objects, postures, etc. In [2.3.19] prolog-like rules are utilised over RDF streams to define patterns for detecting complex situations. In [2.3.14], SWRL rules are used to inference complex temporal interval relations and assertions of new activities, based on the notions of time slices and fluents [2.3.21][2.3.22]. In [2.3.23][2.3.24] the ontological rule-based reasoning is performed by SWRL and SQWRL [2.3.25] to infer inconsistencies between monitored status and scheduled status, e.g. lying in an inappropriate location. Finally, in [2.3.26] a personalized architecture for smart phones is proposed to support PwD undertaking ADLs as they move from one environment to another. The reasoning engine takes as input the models of user preferences, activities and the context and provides customized support in terms of personalized rules that are used to select user preferences based on the current context.

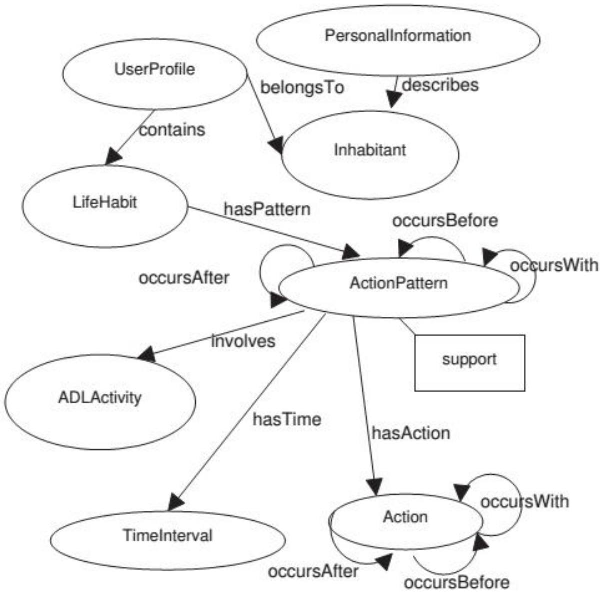


Figure 2.3.1: The behaviour ontological model [2.3.27]

The aforementioned hybrid approaches, however, define ontologies for the representation of basic activity-related information, suppressing the modelling of other intrinsic behavioural aspects, such as the modelling of repeated group of activities over a period of time (routines/habits) or the association of descriptive contexts to domain activities, e.g., the normal duration or the frequency of occurrence of an activity. The Behavioural Ontology in [2.3.27] aims to capture concepts and features of life habits, i.e., long-term behaviours. More specifically, the ontology (Figure 2.3.1) allows capturing repetitive patterns of behaviour and associated semantic information, such as, information about the primitive actions and patterns of primitive actions that characterise a life habit. The life habits encode information about the different ways an individual performs ADL activities and they are identified by analysing the

ADL data of the activity logs [2.3.28], such as the sensor activation sequences, as well as statistical properties of each pattern's occurrence. Towards this end, the notion of *support* is defined as the proportion of the number of times a pattern occurs in the activity log for a particular ADL given the total number of occurrence of all other patterns for that activity.

In [2.3.29], an ontology-based model for capturing user preferences and behaviour routine in the context of a ubiquitous environment is presented. To this end, two OWL ontologies have been created: the so called spatio-temporal ontology of user preference (STOUP) to represent user beliefs, desires, and intentions related to different times and places, and the spatio-temporal ontology of user routine ontology (STOUR) which allows expressing the recurrence of activities and respective locations that comprise the routine. As depicted in Figure 2.3.2, the STOUP ontology allows expressing positive or negative preferences, which are related to a certain time and place, and can also be associated with what the user (agent) is doing, and/or what they intend to do next. The `Preference` class is further specialised into three subclasses, namely `ResourceInterest`, `EnvironmentDesire` and `IntentionalControlCommand`, allowing a preference to refer to resources of interest (e.g. TV channel), environmental desires (e.g. light intensity) and operations (e.g. rolling down the curtain) respectively.

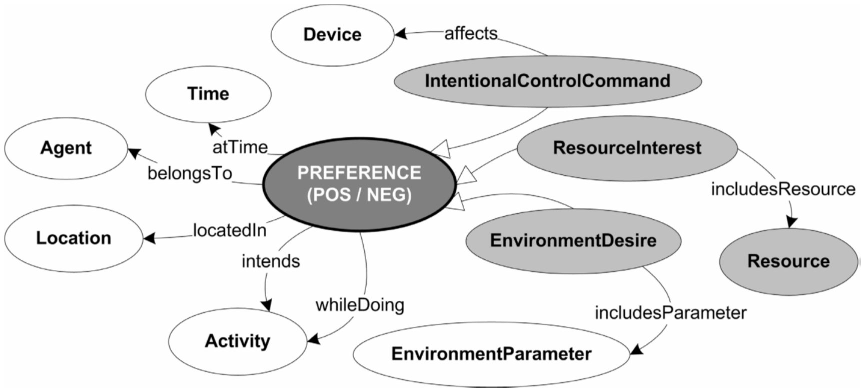


Figure 2.3.2: Spatio-Temporal Ontology of User Preference (STOUP).

In turn, the STOUR ontology (Figure 2.3.3) allows expressing daily, weekly and monthly routines, in the form of ordered sequences of user activities and/or locations, along with their associated expected time interval. The three specialisation classes of the `Routine` class allow expressing the different types of routines, while respective constituent `RoutineItems` can be included through `includesItem` property assertions. Two temporal properties, namely `isAfter` and `isBefore`, are used to capture ordering constraints among the routine items, while user activity, location and time information pertaining to each routine item is captured through respective properties.

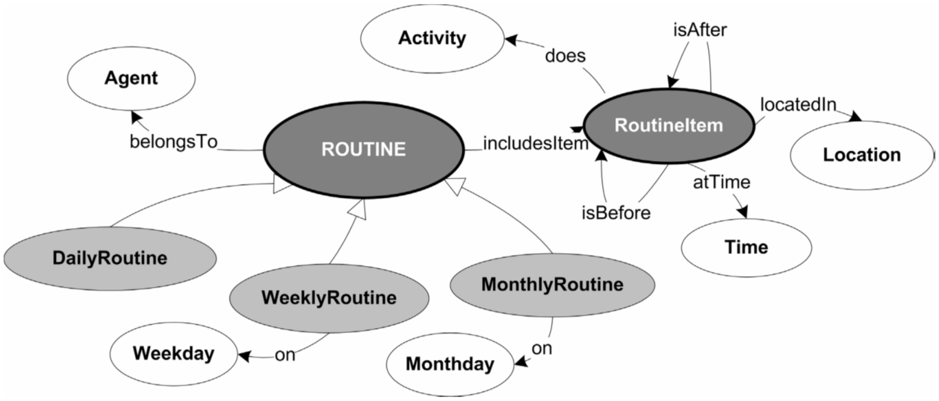


Figure 2.3.3: Spatio-Temporal Ontology of User Routine (STOUR).

In [2.3.30] an ontology-based model is presented for daily activity recognition and detection of behaviour abnormalities in a smart home environment for supervising elderly people. Four types of activity abnormalities are considered: i) duration abnormalities, where there is a discrepancy between the actual duration of the activity and its usual duration, ii) context abnormalities, denoting that an activity is performed in the wrong location, iii) unusual time abnormalities, capturing situations where an activity is performed in an unusual time of the day, and iv) order abnormalities that correspond to activities performed in an order other than the usual one. To support the inference of the aforementioned abnormalities, a domain activity ontology is proposed to model the daily programme of the supervised person (Figure 2.3.4). As illustrated, each activity can be associated with the set of its constituent sub-activities (e.g. reading is composed of walking, reaching the bookshelf, taking a book, walking again and sitting), the set of complex activities that the considered activity may be part of, context information capturing the typical locations where an activity is performed, the typical duration of an activity, its goal, as well as dependencies in the form of temporal sequences involving the activity (e.g. resting takes place after lunch). To reason about activities and abnormalities, the ontology model is translated into a stochastic context-free grammar and tree-parsing.

However, there is still no consensual behaviour ontology in the literature that may be broadly reused to conceptually describe activity patterns and the preferred ways of individuals of doing activities, e.g. routines and habits. A major obstacle to ontology sharing and reuse appears with ontologies in which the intended semantics is captured by the implementation, rather than the axiomatisation [2.3.31], such as in the hybrid approaches, where the semantics is encapsulated in rules, rather than in the domain models. A prominent example is the assertion of new named individuals for representing inferred complex activities, e.g., assert a tea preparation instance that is inferred on the basis of heat water and use tea bag instances. Thus, applications that share similar purpose and scope cannot directly avail of existing ontologies, unless specific implementation details are made available.

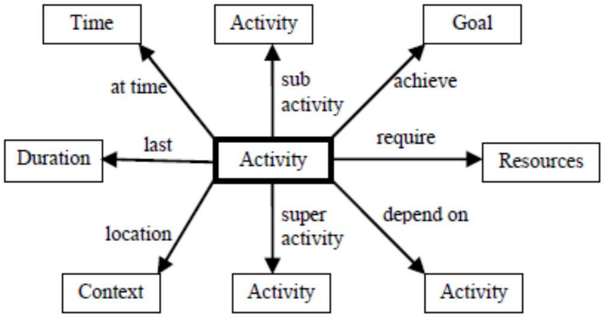


Figure 2.3.4: Activity model from daily programme ontology.

Moreover, the existing behaviour models provide the vocabulary for the representation of asserted relations, e.g., sequence of activities, without capturing the structure and semantics of the respective behaviours. For example, the ontology in Figure 2.3.1 is used to represent action patterns, that is, the temporal relations among the ADL instances that have been detected from the activity logs and characterise a life habit. The ontology, however, falls short to provide reusable descriptions of a life habit, e.g., the domain activity types that are involved, or to associate activity classes with descriptive contexts, e.g. to define the frequency of occurrence of certain activities on a daily or weekly basis.

Finally, due to individual preferences and/or limitations, the performance of ADLs may vary and in some cases, even the same individual may carry out an activity in various ways and sequences. Therefore, there is a need to associate behaviours with multiple descriptive contexts, since different situations merit different distinctions. The existing behaviour ontologies are not flexible enough to represent and manage the different ways (views) of performing activities.

To promote a well-defined description of behaviours and achieve a better degree of knowledge sharing and reuse, we developed an ontology model for behaviour patterns, where the different patterns are represented as specialized instantiations of the descriptions and situations (DnS) ontology pattern [2.3.32] that is part of DOLCE+DnS Ultralite. The developed patterns treat domain classes as instances to allow property assertions to be made among activity types (meta-patterns). In that way, they enable the representation of contextualised views on behaviours, and afford reusable pieces of knowledge that cannot otherwise be directly expressed by the standard ontology semantics.

Several ontologies have been explored as means to capture meta-knowledge in a declarative way. In [2.3.33], a top-level ontology is proposed to model the semantics common to all dimensions of an information space, i.e., levels of granularity, conflicting and overlapping relationships that can be used to evaluate and compare concepts and terms of the ontologies built upon them. In [2.3.34], an ontology-based framework, based on the Event-Condition-Action (ECA) pattern, is presented in order to integrate heterogeneous semantic web services via rule definition. In [2.3.35], an ontology is used to model different types of event rules in order to enable automatic service discovery, while in [2.3.36], a Rule Management Ontology is presented to support the representation of event-based rules that trigger specific actions in a context-aware recommender system. Similar to [2.3.36], we aim to promote reusable and interoperable contextual activity models. However, unlike [2.3.36] that focuses on the definition of a vocabulary representing event-based rules, we use the DnS ontology pattern to formalise high-level behaviour descriptions. As such, the underlying semantics of the behaviour patterns can be reused in already existing frameworks for behaviour modelling and processing.

References

[2.3.1] C. Bettini, O. Brdiczka, K. Henricksen, J. Indulska, D. Nicklas, A. Ranganathan, D. Riboni, "*A survey of context modelling and reasoning techniques*" Pervasive Mob. Comput., vol. 6, no. 2, p. 161–180, Apr. 2010.

[2.3.2] K.L. Skillen, L. Chen, C.D. Nugent, M.P. Donnelly, W. Burns, I. Solheim, "*Ontological user profile modeling for context-aware application personalization*", 6th international conference on Ubiquitous Computing and Ambient Intelligence (UCAmI'12), p. 261-268, 2012

[2.3.3] B. C. Grau, I. Horrocks, B. Motik, B. Parsia, P. PatelSchneider, U. Sattler., "*OWL 2: The Next Step for OWL*" Web Semantics: Science, Services and Agents on the World Wide Web, vol. 6, no. 4, p. 309–322, October 2008.

[2.3.4] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, P. F. Patel-Schneider, Eds., "*The Description Logic Handbook: Theory, Implementation, and Applications*", Cambridge University Press, 2003.

[2.3.5] M.Y. Vardi, "*Why is modal logic so robustly decidable?*", DIMACS Workshop on Descriptive Complexity and Finite Models, DIMACS Series in Discrete Mathematics and Theoretical Computer Science. American Mathematical Society, p. 149–184, Jan. 1996.

[2.3.6] D. Riboni, C. Bettini, "*Cosar: hybrid reasoning for context-aware activity recognition*", Personal Ubiquitous Comput. 15(3), p. 271-289, Mar 2011

[2.3.7] D. Riboni, C. Bettini, "*OWL 2 modeling and reasoning with complex human activities*", Pervasive and Mobile Computing 7(3), p. 379-395, 2011

[2.3.8] L. Chen, C.D. Nugent, "*Ontology-based activity recognition in intelligent pervasive environments*", International Journal of Web Information Systems 5(4), p. 410-430 2009

[2.3.9] L. Chen, C.D. Nugent, H. W, "*A knowledge-driven approach to activity recognition in smart homes*", IEEE Trans. on Knowl. and Data Eng. 24(6), p. 961-974, Jun, 2012

[2.3.10] D. Riboni, L. Pareschi, L. Radaelli, C. Bettini, "*Is ontology-based activity recognition really effective?*", Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on. p. 427-431, March, 2011

[2.3.11] B. Motik, G.B. Cuenca, U. Sattler, "*Structured objects in OWL: representation and reasoning*", Proceedings of the 17th international conference on World Wide Web (WWW '08). p. 555-564, 2008

[2.3.12] B. Motik, U. Sattler, R. Studer, "*Query Answering for OWL-DL with rules*", J. Web Sem., 3(1), p.41–60, 2005.

[2.3.13] I. Horrocks, P.F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, M. Dean, "*SWRL: A Semantic Web Rule Language Combining OWL and RuleML*", Technical report, National Research Council of Canada, Network Inference, and Stanford University, May 2004.

[2.3.14] G. Okeyo, L. Chen, H. Wang, S. Roy, "*A Hybrid Ontological and Temporal Approach for Composite Activity Modelling*", IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), p. 1763-1770, 25-27 June, 2012

[2.3.15] M. Koutraki, V. Efthymiou, G. Antoniou, "*S-creta: Smart classroom real-time assistance*", Ambient Intelligence - Software and Applications, Advances in Intelligent and Soft Computing, vol. 153, p. 67-74, 2012

[2.3.16] K. Wongpatikaseree, M. Ikeda, M. Buranarach, T. Supnithi, A.O. Lim, Y. Tan, "*Activity Recognition Using Context-Aware Infrastructure Ontology in Smart Home Domain*". Seventh International Conference on Knowledge, Information and Creativity Support Systems (KICSS '12), p. 50-57, 2012

[2.3.17] K. Teymourian, M. Rohde, A. Paschke, "*Fusion of background knowledge and streams of events*", *6th ACM International Conference on Distributed Event-Based Systems* (DEBS '12), p. 302-313, 2012

[2.3.18] A. Bolles, M. Grawunder, J. Jacobi, "*Streaming SPARQL extending SPARQL to process data streams*", 5th European semantic web conference on The semantic web: research and applications, p. 448-462, 2008

[2.3.19] D. Anicic, S. Rudolph, P. Fodor, N. Stojanovic, "*Stream Reasoning and Complex Event Processing in ETALIS*", Semantic Web Journal, Special Issue: Semantic Web Tools and Systems, 2012.

[2.3.20] D.F Barbieri, D. Braga, S. Ceri, E.D. Valle, M. Grossniklaus, "*Querying RDF streams with C-SPARQL*", SIGMOD Rec. 39(1), p. 20-26, Sep 2010.

[2.3.21] C. Welty, R. Fikes, "*A Reusable Ontology for Fluents in OWL*", Fourth Int. Conf. on Formal Ontology in Information Systems, p. 226-236, 2006.

[2.3.22] S. Batsakis, E.G.M. Petrakis, "*SOWL: A Framework for Handling Spatiotemporal Information in OWL 2.0*", 5th Int. Symp. RuleML 2011, p. 242-249, 2011

[2.3.23] S. Zhang, P. McCullagh, C. Nugent, H. Zheng, "*An Ontology-Based Context-aware Approach for Behaviour Analysis*", Activity Recognition in Pervasive Intelligent Environments, Atlantis Press, p. 127-148, 2011

[2.3.24] S. Zhang, P. McCullagh, C. Nugent, H. Zheng, N. Black, "*An ontological approach for context-aware reminders in assisted living' behavior simulation*", 11th international conference on Artificial neural networks conference on Advances in computational intelligence - Volume Part II (IWANN'11), p. 677-684, 2011

[2.3.25] M. O'Connor, A. Das, "*SQWRL: a query language for OWL*", Fifth International Workshop on OWL: Experiences and Directions (OWLED), 2009

[2.3.26] K. Skillen, L. Chen, C.D. Nugent, M.P. Donnelly, I. Solheim, "*A user profile ontology based approach for assisting people with dementia in mobile environments*", Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE , p. 6390 - 6393, Aug. 2012

[2.3.27] G. Okeyo , L. Chen , H. Wang , R. Sterritt, "*Ontology-Based Learning Framework for Activity Assistance in an Adaptive Smart Home*", Activity Recognition in Pervasive Intelligent Environments, Atlantis Ambient and Pervasive Intelligence, p. 237-263, 2011

[2.3.28] G. Okeyo, L. Chen, H. Wang, R. Sterritt, "*Ontology-enabled activity learning and model evolution in smart homes*", 7th international conference on Ubiquitous intelligence and computing (UIC'10), p. 67-82, 2010

[2.3.29] K. A. P. Ngoc, Y. K. Lee, S. Lee, *"OWL-Based User Preference and Behavior Routine Ontology for Ubiquitous System"*, On the Move for Meaningful Internet Systems Conference (OTM, Part II), p. 1615-1622, 2005

[2.3.30] I. Mocanu, A. M. Florea, *"A Model for Activity Recognition and Emergency Detection in Smart Environments"*, 1st International Conference on Ambient Computing, Applications, Services and Technologies (AMBIENT), Barcelona, Spain, p. 13-19, Oct. 2011

[2.3.31] M. Grüninger, "Ontology Repositories Make a World of Difference", Theory, Practice, and Applications of Rules on the Web (RuleML 2013), p. 12-12, 2013

[2.3.32] A. Gangemi, P. Mika, *"Understanding the semantic web through descriptions and situations"*, International Conference on Ontologies, Databases and Applications of Semantics. p. 689-706, 2003

[2.3.33] J. Ye, G. Stevenson, S. Dobson, "*A top-level ontology for smart environments*", Pervasive and Mobile Computing 7(3), p. 359 – 378, 2011

[2.3.34] W. May, J.J Alferes, R. Amador, "*An ontology- and resources-based approach to evolution and reactivity in the semantic web*", In: OTM Conferences (2), p. 1553 – 1570, 2005

[2.3.35] V. Beltran, K. Arabshian, H. Schulzrinne, "*Ontology-based user-defined rules and context-aware service composition system*", In: ESWC Workshops. p. 139-155, 2011

[2.3.36] J. Debattista, S. Scerri, I. Rivera, S. Handschuh, "*Ontology-based rules for recommender systems*", In: SeRSy, p. 4960, 2012

.

3 Unsupervised Activity Recognition using Fixed Cameras

3.1 Introduction

The complete framework that we propose is able to recognize long-term (hours) activities in an unsupervised manner and can be used in unstructured scenes. It uses contextual information to automatically create an intermediate structure of action primitives in order to build a hierarchical activity model that characterizes an activity. This is done in several steps: (a) long-term videos are processed in order to obtain information about the movement (features) about an observed person (i.e. global positions and the motion of his/her body parts). (b) Features are used to learn the scene regions (what we call topology) in multi-resolution levels. (c) Features and scene regions are spatially and temporally fused to build primitive events which represent an action primitive, such as passing from one region to another. (d) Based on the primitive events in different resolution levels, activities are modelled in a hierarchical. (e) Recognition is achieved by comparing similarity between models of activity. These steps are explained in detail in the following section.

3.2 Discovering activities from video features

3.2.1 Low-level Processing: Feature Extraction

At each particular time-stamp of a long time video, our framework extracts a set of space-time trajectory features describing the global position of an observed person and the motion of his/her body parts. The information about the motion of the person is gathered in a chunk, that we call Feature Chunks (FC), in order to represent the motion in the scene. The information is obtained after decomposing the video into short sequences of images (i.e. video chunks) based on significant changes of human motion (e.g., in speed). Next, we are going to describe how FC are extracted using the input data from 2D and RGB-D cameras.

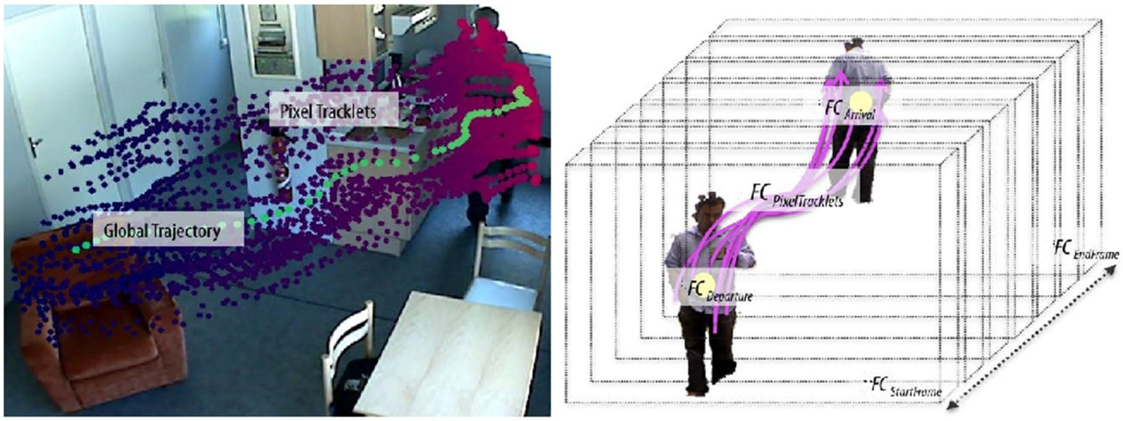
Feature Extraction from 2D Cameras

The position of a person is estimated by using a set of tracklets, which is computed for each video chunk by tracking particular corner points. First, 500 corner points [3.4] are randomly initialized and tracked over time using KLT [3.2]. Second, we compute 4 clusters (k-means) of the points with respect to their speed and position, representing static, slow, medium and fast motion. Finally, we compute the global position p_t of the person at time t, by averaging the centroids of the 3 point clusters (i.e. slow, medium and fast motion).

Due to noise in images p_t can be unreliable. Therefore, we obtain a smoothed global position $\widetilde{p}_t$ by applying a Kalman filter K_1 to p_t in combination with the last n_s smoothed positions:

$$\widetilde{p}_t = \frac{1}{n_s + 1}\left(p_t + \sum K_1(\widetilde{p}_{t-i})\right) \qquad (3.1)$$

The sequence of $\{\tilde{p}_t\}$ represents the global trajectory which is represented in Figure 3.1-(a) by green points.



(a) Global trajectory and Pixel tracklets

(b) FC features

Figure 3.1: Global trajectories (green) & Pixel Tracklets (purple to pink)

We compute the person speed s_t at time t as the difference of the person position at time t and $t-1$. Similarly, we compute a smoothed speed $\tilde{s}_t$ by applying a Kalman filter K_2 to s_t , in combination with the last n_s smoothed speeds:

$$\tilde{s}_t = \frac{1}{n_s+1}\left(s_t + \sum K_2\left(\tilde{s}_{t-i}\right)\right) \tag{3.2}$$

Finally, the video is decomposed into video chunks by comparing $\tilde{s}_t$ with a threshold.

Consequently, each video chunk is associated with a Perceptual Feature Chunks with the following attributes: $Departure_{FC}$, $Arrival_{FC}$ which are two Gaussian distributions characterizing the person positions at the beginning and end of the video chunk. The mean and standard deviation (μ,σ) of the position distributions are computed using the first (or last) n_g points of the global trajectory. $StartFrame_{FC}$, $EndFrame_{FC}$ represent the first and last frame number of the video chunk. $PixelTracklets_{FC}$ are the pixel-based tracklets used to calculate the agent global trajectory. An example of $PixelTracklets_{FC}$ (pink to purple) of a person moving from the armchair to the kitchen is represented in Figure 3.1-(a). An illustration of the feature chunks attributes can be found in Figure 3.1-(b). The feature chunks enable to collect the necessary information for activity understanding and to avoid expensive computational time, especially for long-term activities.

Feature Extraction from RGB-D Cameras

The position of the person in RGB-D camera view is estimated by using the person detection algorithm explained in [3.1]. Using this algorithm, the centre point of the bounding box that covers the people in the scene is estimated. Simply, at each frame we use this algorithm to detect the centre point of the person in the scene and create a trajectory by concatenating the centre points.

Unlike what we do for 2D cameras, since we represent a person using one point, we do not need to cluster the points by their speed. We decompose the video into chunks in every two frames and set $Departure_{FC}$, $Arrival_{FC}$ parameters as trajectory points at those particular frames. Again, $StartFrame_{FC}$ and $EndFrame_{FC}$ represent the first and last frame number of the video chunk.

Since we do not extract local motion of the person, we do not use the parameter $PixelTracklets_{FC}$ in FCs for RGB-D cameras. In future, we are going to use an extended version of our person detection algorithm that will detect body parts such as head, hands, feet, etc. and record the local motion in $PixelTracklets_{FC}$ parameter.

3.2.2 Topology Learning

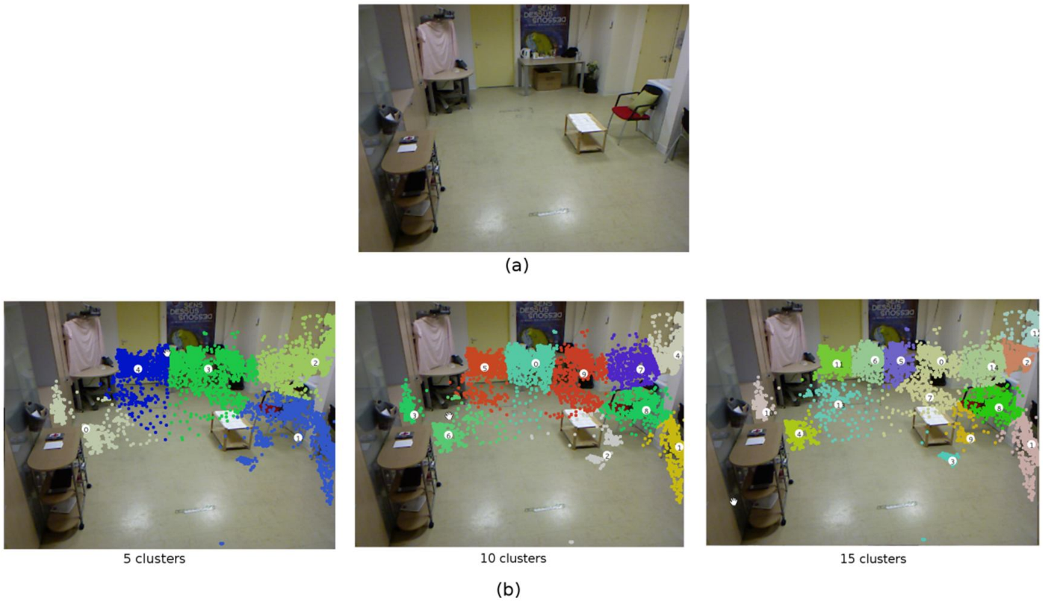
When a tracked person performs activities, he/she interacts with many objects that can be represented by fixed regions (e.g. the person interacts with the kitchen to prepare meal). We name each set of scene regions a topology (or contextual information) and learn each topology by clustering trajectory points ($\{\widetilde{p}_t\}$).

To learn a topology, we use the Perceptual Feature Chunks associated to one or several people performing activities in the same scene at various time. From this set of sequences, we extract a set of points $Points_{Seq}$ of the $Departure_{FC}$ and $Arrival_{FC}$ of all videos.

$$Points_{Seq} = \{Departure_{FC}(\mu)\}\text{U}\{Arrival_{FC}(\mu)\} \qquad (3.3)$$

We perform k-means clustering [3.3] over $Points_{Seq}$. The number of clusters represents the level of granularity of the topology, where lower numbers imply smaller number of regions that are wider. Each cluster defines a $SceneRegion(SR)$. We denote a topology at level $l$associated with k clusters as $T_l = \{SR_0^l, \dots, SR_{k-1}^l\}$.

We represent a scene model as a set of topologies of different resolution levels. We propose for building this scene model to calculate 3 levels of topologies that correspond to 5, 10 and 15 clusters. Figure 3.2 describes the scene model obtained by a clustering procedure in a hospital room for our dataset described in Section 3.3.


(a)

5 clusters

10 clusters

15 clusters

(b)

Figure 3.2: (a) Initial scene (b) Scene model example with l = 1, 2 and 3 obtained by k-means clustering (k = 5, 10 and 15, respectively)

3.2.3 Primitive Events

We propose an intermediate layer called $PrimitiveEvent(PE)$ that enables to link gradually the extracted features (low-level information) from images to the semantic interpretation of the scene (high-level information). A PE is the event characterizing a Feature Chunks (Section 3.2.1) over a single topology (Section 3.2.2). For each agent, a sequence of PE is built using the sequence of Feature Chunks and a topology T_l. In practice, we build 3 sequences of PE (for three levels of topology, $(l = 1,2,3)$ for a single input video.

A PE consists of 4 attributes. We will describe the two of them: $Transition_{PE}$ and $LocalDynamics_{PE}$ are described in the following subsections.

The $Transition_{PE}$

It describes the movement of an agent over the scene by extracting the transition information performed between the learned scene regions SR_i^l at one level, l.

The $Transition_{PE}$ is represented as a directed pair of regions:

$$Transition_{PE} = (StartRegion \rightarrow EndRegion) \tag{3.4}$$

where $StartRegion$ and $EndRegion$ are the labels of the nearest SR_i^l (ith scene region from T_l) to the $Departure_{FC}(\mu)$ and $Arrival_{FC}(\mu)$ positions.

The $LocalDynamics_{PE}$

The $Transition_{PE}$ can only describe the agent global motion while he/she performs an activity over the scene (moving from one region to another one or staying in a region). To be able to model finer activities (low-level activities), we compute the $LocalDynamics_{PE}$ attribute that contains finer features (point tracklets) on the movement of the agent body parts (hands, arms, torso, etc.).

The $LocalDynamics_{PE}$ are obtained by clustering the $PixelTracklets_{FC}$ (Section 3.2.1). For clustering, we use the mean-shift algorithm [3.5]. In the literature, the methods for tuning the bandwidth of the mean-shift algorithm are not appropriate to compute a finer description of the local motion. Thus, we adapt the mean-shift bandwidth automatically as a function of the agent global position:

$$h = \left\| Departure_{FC}(\mu) - Arrival_{FC}(\mu) \right\| \tag{3.5}$$

where h is the bandwidth window. Figure 3.3 illustrates five examples of the computed $LocalDynamics_{PE}$ (green) from the clustering of the $PixelTracklets_{FC}$ (pink) associated to the following movements: arms up, arms down, join hands, bend down and stretch up. It can be noticed how the local dynamics (green tracklets) can capture five activities while the person remains at the same location.

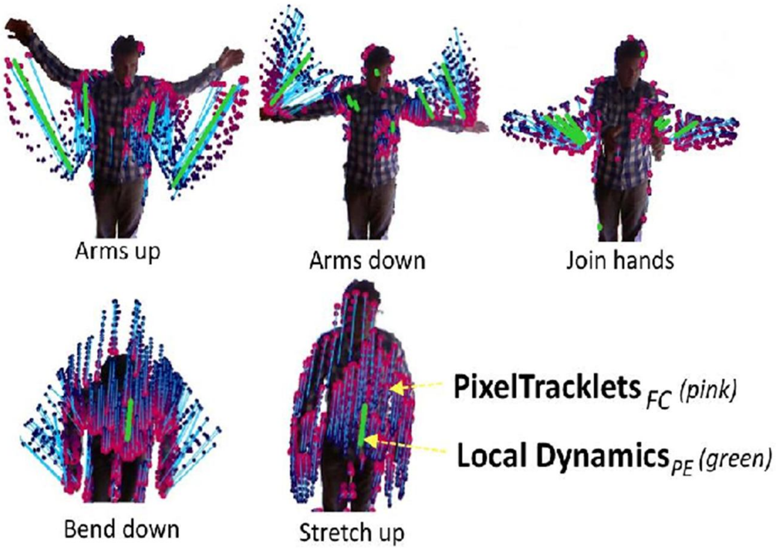


Figure 3.3: Example of the abstraction of $PixelTracklets_{FC}$ into $LocalDynamics_{PE}$.

3.2.4 Building the Hierarchical Activity Model

Activity Discovery

The sequences of primitive events are very informative about the activity occurring in the video (See Figure 3.4). However, a $PrimitiveEvent$ can only describe a snapshot of the agent's motion. To provide more meaning, a better representation of the discovered activity is needed.

If a person is staying in a region for a certain time, we need to fuse the sequences of PE to obtain one global activity corresponding to all the time he/she stayed in the region. Another kind of activity occurs when the agent is moving from one region to another one. Therefore, we consider two patterns $Change$ and $Stay$ to describe the two types of activity as follows:

- The $Stay$ pattern characterizes an activity occurring within a single topology region like "at.region.P", and it is defined as a maximal sub-sequence of PE with the same $Transition_{PE}$:

$$Stay_{P-P} = (P \to P) \tag{3.6}$$

- The $Change$ pattern describes the transition of the agent between regions like "changing.from.P.to.Q" which is composed of a single PE:

$$Change_{P-Q} = (P \to Q), \quad P \neq Q \tag{3.7}$$

We define a discovered activity (DA) at a level l as an extracted $Stay_{P-P}$ or $Change_{P-Q}$ pattern:

$$DA^{l}_{P-Q} = Stay_{P-P} \mid Change_{P-Q} \tag{3.8}$$

The process of activity discovery is performed over the three granularity levels ($l = 1, 2, 3$) using the three sequences of PE for three levels of topology. Therefore, based on the hierarchy of the scene regions, the discovered activities are also classified to coarse, medium and fine and each discovered activity is a sub-activity of an activity at a coarser resolution. Figure 3.4 presents an example for discovered activities DAs extracted with the $Change$ and $Stay$ patterns at multiple resolutions.

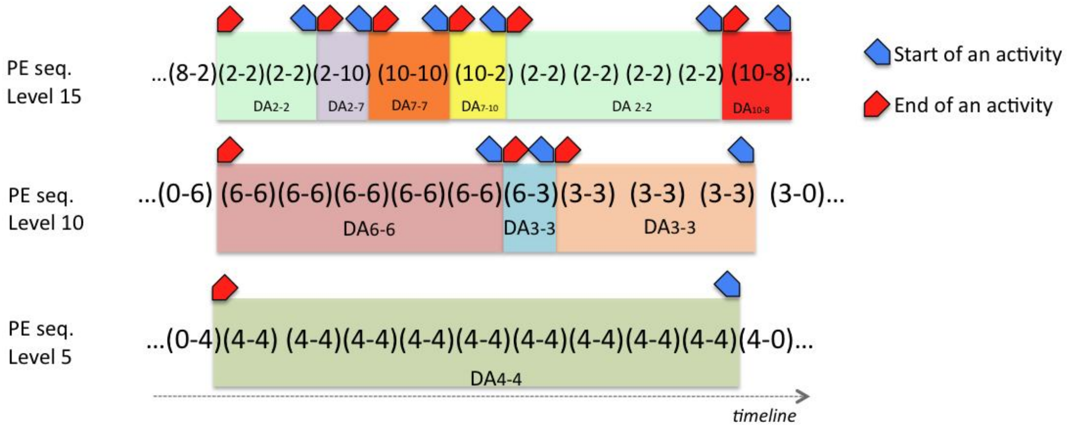


Figure 3.4: Example of discovered activities DAs (colored segments) extracted with the $Change$ and $Stay$ patterns at multiple resolutions.

In the following sections, we replace $P-Q$ and $P-P$ by the index s that represents the semantic of an activity which are mapped to colours on the graphical interface to categorize the activities in the video. Figure 3.5 shows the coloured segments representing the discovered activities at three levels of resolution. The same colour corresponds to the same activity at each resolution level.

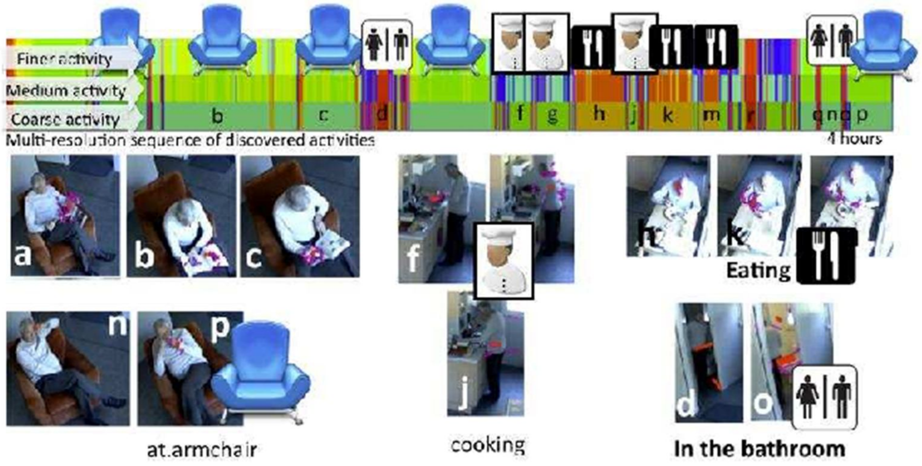


Figure 3.5: Example of discovered activities (coloured segments) for 4 hours video of one person performing everyday activities

Definition of Activity Models

We represent the model of an activity as a tree of nodes that is obtained by merging the set of $\left\{DA_s^{l=1,2,3}\right\}$ (s is the semantics of the activity) and has a hierarchical structure based on the three levels of granularity (i.e. $\left\{N^{l=1}, \left\{N_i^{l=2}\right\}_{1\leq i\leq n}, \left\{N_j^{l=3}\right\}_{1\leq j\leq m}\right\}$). The tree of nodes represents how different activities and sub-activities are connected to each other thanks to a set of $attributes$ and $sub-attributes$ obtained from the properties such as type, duration, etc. In other words, a node N is characterized by $attributes$ and $sub-attributes$ parameters:

- The $attributes$ is a set of parameters over the DA_s at the current level l that characterizes the node N^l.
- The $sub-attributes$ constitutes the set of parameters that characterizes the attributes of the sub nodes N_i^{l+1}, where i is the index of the child node of N^l.

Learning of Activity Models

For a selected instances of the same discovered activities DA_s^l (e.g. $s =$ "cooking"), we learn the model of activity by constructing a tree of nodes where each node of level l is built from the set of discovered activities that are at the same resolution level l, $\left\{DA_{s_1}^l, DA_{s_2}^l, \ldots, DA_{s_n}^l\right\}$

where $s_1, s_2, \dots, s_n$ are parts of s (i.e. sub-activities of cooking). An example of the constructing process of a tree of nodes from three sequences of discovered activities classified from the coarser to the finer one is illustrated in Figure 3.5-(a). We construct an independent model for each type of discovered activity. In the following subsections, we describe the parameters of $attributes$ and $sub - attributes$.

The $attributes$ of a node: For a node N^l, we define 3 attributes to describe temporal and spatial properties of a node:

- $Type$: it is adopted from the DA_s composing a node. For a node N, $Type_N = Type_{DA_s}$
- $Instances$: the amount of training instance of activities composing a node.
- $Duration$: a Gaussian distribution $N(\mu_d, \sigma_d^2)$ describing the temporal duration of the training instances.
- $HistogramofLocalDynamics$ $H(\theta)$: is a histogram that characterizes the distance and the angle of local motion (we discretize the angles to 8 bins (see Figure 3.6-(b)) that describes a histogram built from the set of discovered activities).

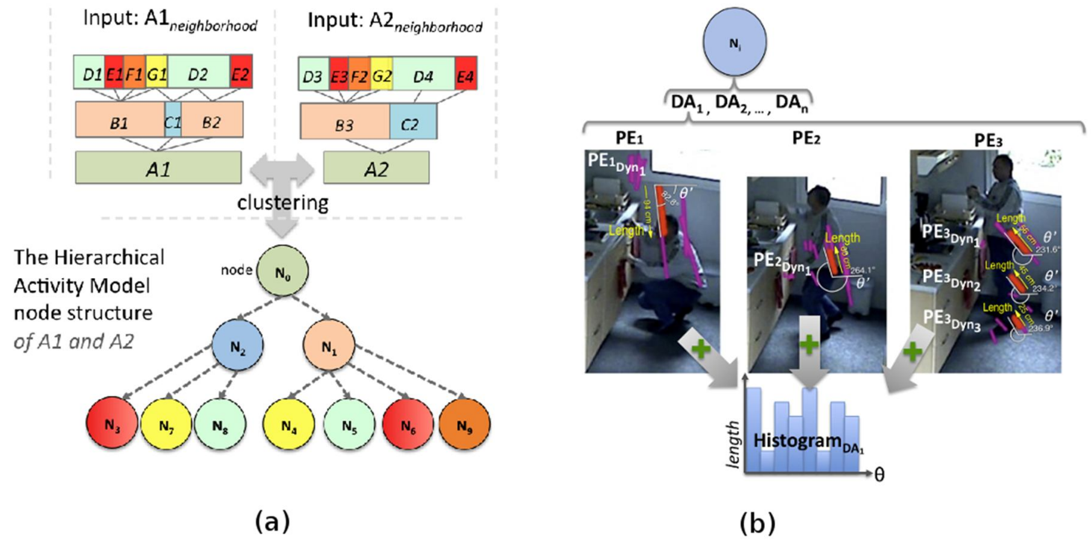


Figure 3.6: (a) Hierarchical Activity Model (HAM) (b) Histogram of Local Dynamics

The $sub - attributes$ of a node: The sub-attributes enable us to get information from the child nodes. To compute the sub-attributes of a node, we use the attributes of its child nodes. For a node N^l, we define two sub-attributes named $mixture_{sub-activity}$ and $time - elapse_{sub-activity}$ which aim at describing two properties of the child nodes N_i^{l+1} of N^l :

1. $mixture_{sub-activity}$: describes the amount of time a child node with the same $Type$ appears. It is represented as a mixture of Gaussians (MOG) mixture of ($\theta_{type}^{mixture}$) with the following parameters:
   - K, is the total number of components (Gaussians), it is equivalent to the number of unique $Types$
   - O, is the total number of discovered activities at level l (DA^{l}).
   - $w_{q=1,\dots,K}$, is the prior probability of the component q^{q}. It is equivalent to the weight of each Gaussian in the MOG. It is computed based on the number of appearances of the nodes with the same $Type$:

$$w_{q} = \frac{\sum \delta\left(Type_{N_{p}^{l+1}}, Type\right)}{O}$$

Then, $\theta_{type}^{mixture} = \sum w_q * N(\mu_q, \sigma_q^2)$ where μ_q is calculated by the training instances of all child nodes with the same $Type$:

$$\mu_{q} = \frac{\sum Instances_{N_{p}^{l+1}} * \delta\left(Type_{N_{p}^{l+1}}, Type\right)}{\sum \delta\left(Type_{N_{p}^{l+1}}, Type\right)}$$

2. $time - elapse_{sub-activity}$: represents the temporal distribution of child nodes. For an activity, it describes the expected temporal duration of its sub-activities. $time - elapse_{sub-activity}$ is also represented by a MOG of ($\theta_{type}^{timeelapse}$). The parameters of $time - elapse_{sub-activity}$ are similar to previous sub-attribute $mixture_{sub-activity}$.

Recognition of Activity Models

For a new unseen video dataset, we aim at recognizing activities in an unsupervised way. The task is achieved by measuring the similarity between reference activity models that are learned for each type of discovered activity using unlabelled training videos and a test activity model that is obtained from the discovered activities of the new video.

First, a new sequence of Feature Chunks is computed for the new video. Second, using three levels of topology learned from training videos, we create new $PrimitiveEvent$. Thereby, $Transition_{PE}$ of new PE are matched with the $Transition_{PE}$ of PE used in training. Third, the activity discovery process is performed with the new PE and a new sequence of discovered activities is computed. Fourth, for each type of discovered activity of the new video, an activity model is built as explained in previous section. Finally, we compute a score between the new model and learned models and threshold it to classify the activity.

To compute a similarity score between two activity models, we define a metric in a recursive manner. At each level of the model, we calculate a similarity score by computing the Euclidean distance between attributes and sub-attributes of the nodes of two models at that level and append the similarity score obtained from the finer level. This recursive procedure gives us the opportunity to have a similarity score at the root node that measure the similarity of the models at all levels.

3.3 Results & Discussion

We have tested our unsupervised activity recognition method using both 2D and RGB-D data. The RGB data of RGB-D camera are used as images from 2D camera. In the experiments, 10 videos from the dataset that include semi-guided daily activities are used. 5 videos are used to build the models for “looking at bus map”, “watering plant”, “preparing tea”, “talking to the phone”, “preparing drugs”, “paying bill”, “watching TV”, and “reading newspaper” actions. In training, the models are constructed in an unsupervised way, and then each model is manually labelled to have semantic descriptions. The remaining five videos are used for testing. In testing, as explained in previous sub-section, a semantic label for each discovered activity of the test video is assigned by comparing with the models built from training videos.

To evaluate the framework, we use the following definitions:

$TP = TruePositive$: Number of activities correctly recognized

$FP = FalsePositive$: Number of activities recognized not appearing in the ground truth.

$FN = FalseNegative$: Number of non-recognized activities.

TPR: true positive rate (also called recall rate or sensitivity in some publications) measures the proportion of actual positives which are correctly identified as such, it is defined as:

$$TPR = \frac{TP}{(TP+FN)} \quad -\text{ higher is better }-$$

FDR: false discovery rate, is analogous to the TPR, it is defined as:

$$FDR = \frac{FP}{(FP+TP)} \quad -\text{ lower is better}-$$

PPV: positive predictive value (equivalent to precision), it is defined as:

$$PPV = \frac{TP}{(TP + FP)}$$ – higher is better–

	Using RGB-D data						Using 2D data					
	TP	FP	FN	TPR%	FDR%	PPV%	TP	FP	FN	TPR%	FDR%	PPV%
watering plant	3	49	0	**100**	94,23	5,77	1	11	2	33,33	91,67	8,333
looking at bus map	8	63	0	**100**	88,73	11,27	5	3	3	62,5	37,5	62,5
talking to the phone	7	15	0	**100**	68,18	31,82	7	14	0	**100**	66,67	33,33
preparing tea	0	16	2	0	100	0	1	17	1	**50**	94,44	5,556
preparing drugs	4	84	0	**100**	95,45	4,55	4	4	0	**100**	50	50
reading newspaper	4	55	2	66,67	93,22	6,78	6	8	0	**100**	57,14	42,86
watching TV	5	48	0	**100**	90,57	9,43	0	7	5	0	100	0
paying bill	4	62	1	**80**	93,94	6,06	4	1	1	**80**	20	80

Table 3.1: The performance of our unsupervised activity recognition method using 2D and RGB-D cameras. Bold values represent the best recognition rate for that action

In Table 3.1, the performance of our approach using 2D and RGB-D data are given. We can see that, using either 2D or RGB-D data, we achieve similar level of performance for "talking to the phone", "preparing drugs" and "paying bill" actions. By using 2D data, we are successful at recognizing "talking to the phone", "preparing drugs", "reading newspaper" and "paying bill". Among them, the first three actions are recognized with 100% true positive rate. By using RGB-D data, we can successfully recognize "watering plant", "looking at the bus map", "talking to the phone", "preparing drugs", "watching TV" and "paying bill". Among these only for "paying bill", we obtain 80% true positive rate. All of the other actions are recognized with 100% true positive rate. Following these results, we can say that by using RGB-D data, we obtain better results compared to using 2D data, especially for "watering plant", "looking at bus map" and "watching TV" actions. On the other hand, for "preparing tea" and "reading newspaper", by using 2D data we achieve better results than using RGB-D data.

One of the main reasons of failures in using 2D data arises from the problems in feature point detection and tracking. Sometimes, the points are detected in wrong places, therefore the global position and local dynamics of the person cannot be obtained accurately. This makes it impossible to represent the motion of the person, thereby causes the method either miss or give false alarms. In the dataset we have used in our experiments, the field of view of the camera is not suitable to observe "watching TV" action. Thus, we obtain 0% true positive rate for this action. Similarly, one of the main reasons of failures in using RGB-D data is the failures in person detection. Occasionally, our algorithm detects people in wrong places. For this reason, either the activities are not missed or wrong activities are recognized (false alarms) because of the person detected in a different place.

It can be observed that there are many false positives in both approaches, especially using RGB-D data. The misleading person detection and tracking triggers this problem. Furthermore, in our dataset, the activity zones for some actions are too close to each other. For instance, the zone of "looking at bus map" and "watering plant" are too close. Also, they are very close to the door in the room. For this reason, sometimes, they are mixed up with each other, therefore, we discover actions that did not happen and this creates false positives. In order to cope with this issue, we are going to improve our person detection and tracking algorithms. In addition, we believe that the local dynamics information we obtain from the movement of the person enables us to distinguish activities even when they are in the same area of the scene. By using larger training data, we plan to improve the activity models in this manner, and thereby decrease the rate of missed activities and false alarms.

3.4 Conclusion

In this section, we propose a complete unsupervised framework for discovering, modelling and recognizing ADL using a fixed camera in an unstructured scene. This framework includes all steps from the low-level processing (person detection and tracking) to the semantic interpretation of the motion in the scene, i.e. "preparing tea". Global and local human features are extracted from RGB and Depth data and they are used to learn all the meaningful areas (topologies) of the scene in an unsupervised way. Combining global and local features with topologies enables us to build primitive events in the video at three levels of resolutions. Based on these steps, we have proposed a new model for representing activities which benefits from the multiple-resolution input: Hierarchical Activity Model.

This framework has been successfully tested for recognizing ADL by experimenting on patients performing daily living activities in a hospital room. Although there are missed activities and false, high true positive rates we achieved in the experimental results show that the framework is a promising system that can automatically discover, learn and recognize ADLs. We believe that our method can be used to study activities in home care applications and to perform fast and reliable statistics that can help doctors to diagnose diseases such as Alzheimer. In future, we are going to work on decreasing false positive and false negative rates by improving our motion and person detection algorithms and by improving our activity models using a larger training data.

References

[3.1] Anh-Tuan Nghjem, Edouard Auvient, Jean Meunier: Head detection using Kinect camera and its application to fall detection, ISSPA 2012: 164-169

[3.2] Bouguet, J Y, Pyramidal implementation of the affine Lucas Kanade feature tracker description of the algorithm, Intel Corporation. 2001

[3.3] MacQueen, J B, Some methods for classification and analysis of multivariate observations, in: Proceedings of the Fifth Berkeley symposium on mathematical statistics and probability, California, USA. p. 14.1967

[3.4] Shi, Jianbo and Tomasi, Carlo, Good features to track, In IEEE CVPR'94, pages 593–600, 1994

[3.5] Subbarao,Raghav and Meer, Peter, Nonlinear Mean Shift over Riemannian Manifolds, International Journal of Computer Vision 84, 1-20. 2009

4 Supervised Activity Pattern Recognition in Wearable Video

This section presents our approach for activity recognition in videos taken form wearable cameras. Our work makes three contributions:

- We demonstrate that visual saliency maps based on geometric-spatio-temporal cues [4.3] are a major benefit for distinguishing the location of active objects in egocentric videos;
- We show that analysing the dynamics of a sequence of active objects & context by means of temporal pyramids [4.1] becomes a suitable paradigm for activity recognition in egocentric videos. However, in this scope, we claim that context can be better described by the output of place recognition module rather than by the outputs of many non-active object detectors as proposed in [4.1].
- We provide experimental evaluations on the "subject's point of view" using a publicly available dataset and demonstrate benefits of our model for activity recognition.

In this section, we provide a brief description of the approach used in our place and object categories recognition algorithms and then present how their outputs are processed by the activity recognition module.

4.1 The Approach

Our activity recognition system takes as inputs the outputs of two proceeding modules in the processing pipeline: a) several *Active Object detectors*, and b) a *Place Recognition* module. Once these two modules are described, we present our particular approach for Activity Recognition.

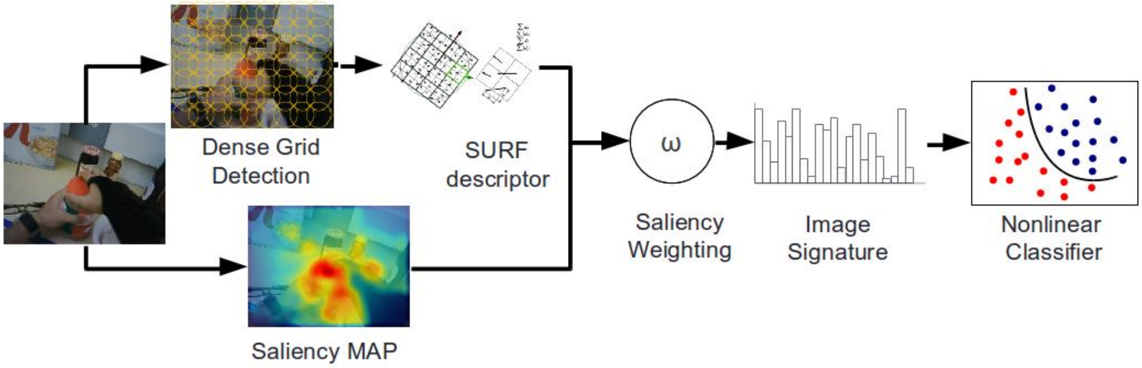


Figure 4.1: Processing pipeline for the saliency-based object recognition in first-person camera videos

4.1.1 Object Recognition

In our system, we employ several detectors of "active objects" (objects either manipulated or observed by the user). From our point of view, identifying these objects becomes a crucial step towards the activity understanding in egocentric daily videos. In general, we consider one individual detector for each object category although, as shown in the processing pipeline presented in Figure 4.1, almost all the processing modules are common for every detector. In practice, the nonlinear classification stage is the only step that is specific to each category. We have built our model on the well-known Bag-of-Words (BoW) paradigm [4.4] and proposed to add saliency masks as a way to provide spatial discrimination to the original Bag-of-Words approach. Hence, for each frame in a video sequence, we extract a set of N SURF descriptors d_n [4.5], using a dense grid of circular local patches. Next, each descriptor d_n is assigned to the most similar word $j = 1 \dots V$ in a visual vocabulary by following a vector-quantization process. The visual vocabulary, computed using a k-means algorithm over a large set of descriptors in the training dataset (about 1M descriptors in our case), has a size of $\mathrm{V} = 4000$ visual words.

In parallel, our system generates a geometric-spatio-temporal saliency map S of the frame with the same dimensions of the image and values in the range $[0,1]$ (the higher the more salient a pixel is). The saliency map computation has been explicitly developed for egocentric vision for wearable camera setting in the framework of the Dem@Care project and the details about the generation of saliency maps can be found in [4.3].

We use this saliency map to weight the influence of each descriptor in the final image signature, so that each bin j of the BoW histogram H is computed using the next equation:

$$H_j = \sum_{n=1}^{N} \alpha_n w_{nj}$$

where the term $w_{nj} = 1$ if the descriptor or region n is quantized to the visual word j in the vocabulary and the weight α_n is defined as the maximum saliency value S found in the circular local region of the dense grid. Finally, the histogram H is L1-normalized in order to produce the final image signature.

Once each image is represented by its weighted histogram of visual words, we use a SVM classifier [4.6] with a nonlinear Chi-Square kernel, which has shown good performances in visual recognition tasks working with normalized histograms as the ones used in the BoW paradigm [4.7]. Using the Platt approximation [4.8], we finally produce posterior probabilistic estimates O_k^t for the occurrence of the object of class k in the frame t.

4.1.2 Place Recognition

In this section we detail the place recognition module. The general framework can be decomposed in three steps. First of all, for each image, a global image descriptor is extracted. We choose the Composed Receptive Field Histograms (CRFH) [4.9] since it was proven to produce good performances for indoor localization estimation [4.10]. Then a non-linear dimensionality reduction method is employed. In our case, we use a Kernel Principal Component Analysis (KPCA) [4.11]. The purpose of this step is two-fold: it reduces the size of the image descriptor, alleviating the computational burden of the rest of the framework, and it provides descriptors on which linear operations can be performed. Finally, based on these features, a linear Support Vector Machine (SVM) [4.6] is applied to perform the place recognition, and the result is regularized using temporal accumulation [4.10].

For the application considered in our work, each video is taken in a different environment. Consequently, our module has to learn generic concepts instead of specific ones as it is usually the case [4.10]. In this context, we need to define concepts both relevant for action recognition and as constrained as possible to obtain better performances. For instance, the concept “stove” has probably less variability and may be more meaningful for action recognition than the concept “kitchen”. This will be discussed in detail in the experimental results section.

Again, following the Platt approximation [4.8], the output of this module is then a vector P_j^t with the probability of a frame t representing the place$j$.

4.1.3 Activity Recognition

Our activity recognition module uses the temporal pyramid of features presented in [4.1], so as to allow exploiting the dynamics of user's behaviour in egocentric videos. However, rather than combining features for active and non-active objects, we study the combination of active objects and places (context) with the aim of modelling activities as sequences of features that involve varying manipulated/observed objects and places (e.g., cooking may involve user's interaction with various utensils whereas cleaning the house might require a user to move around different places of the house).

In particular, for each frame t being analyzed, we consider a temporal neighbourhood Ω_t corresponding to the interval $[-\frac{\Delta}{2},\frac{\Delta}{2}]$ around t. This interval is then iteratively partitioned into two sub-segments following a pyramid approach, so that at each level $\mathrm{l} = 0 \ldots \mathrm{L} - 1$ the pyramid contains 2^l sub-segments. Hence, the final feature of a pyramid with L levels is defined as:

$$F_t = \left[F_t^{0,1} \ldots, F_t^{l,1} \ldots, F_t^{l,2^l} \ldots, F_t^{L-1,2^{L-1}}\right]$$

where $F_t^{l,m}$ represents the feature associated to the sub-segment m in the level l of the pyramid and is computed as:

$$F_t^{l,m} = \frac{2^{l-1}}{\Delta} \sum_{s\epsilon\Omega_{tm}^l} f_s$$

where Ω_{tm}^l represents the m temporal neighbourhood of the frame t in the level l of the pyramid and f_s is the feature computed at frame s in the video. In the experimental section, we will assess the performance of our approach using the outputs of K object detectors $[\mathrm{O}_1^\mathrm{s} \ldots \mathrm{O}_\mathrm{K}^\mathrm{s}]$, the outputs of J place detectors $\mathrm{P}_1^\mathrm{s} \ldots \mathrm{P}_\mathrm{J}^\mathrm{s}$, or the concatenation of both, as features f_s.

In our experiments, we have used a sliding window method with a fixed window of size Δ, parameter that is later studied in the experimental results section, and a pyramid with $L = 2$ levels. Finally, the temporal feature pyramid has been used as input for a linear multiclass SVM in charge of deciding the most likely action for each frame.

4.2 Results

4.2.1 Experimental Setup

We have assessed our model in a publicly available ADL dataset [4.1], which contains videos captured by a chest-mounted GoPro camera, which is used in Dem@Care project as well on 20 users performing various daily activities at their homes. This dataset was already annotated for 44 object-categories and 18 activities of interest (see Figure 4.2) and we have additionally labelled 5 rooms and 7 places of interest.

This dataset is very challenging since both the environment and the object instances are completely different for each user, thus leading to an unconstrained scenario. Hence, and due to the hierarchical nature of the activity recognition process, we have trained every module following a leave-k-out procedure ($k = 4$ in our approach). This approach allows us to provide real testing results in object and place recognition for every user, so that the whole set can be later used for activity recognition. Furthermore, for activity recognition, the first 6 users have been used to cross-validate the C parameter of the linear SVM, whereas the remainder ones (7-20) have been used to train and test the models following a leave-1-out approach.

4.2.2 Object recognition results

Figure 4.3 shows the per-category and average results achieved by our active object detection approach in terms of Average Precision (AP). The mean AP of our approach is 0.11 but, as can be noticed from the figure, the performance notably differs from one class to another. Main errors in classification are due to various reasons: a) a high degree of intra-class variation between instances of objects found at different homes, what leads to poor recognition rates (e.g. bed clothes or shoes show large variations in their appearance), b) some objects are too small to be correctly detected (dent floss, pills, etc.), and c) for some objects that theoretically show a lower degree of intra-class variation (TV, microwave), performance is lower than expected since it is very hard for a detector to distinguish when they can be considered as “active” in the scene (e.g. a user just faces a “tv remote” or a “laptop” when using them, whereas the TV or the microwave are more likely to appear in the field of view even when they are not ‘active’ for the user).

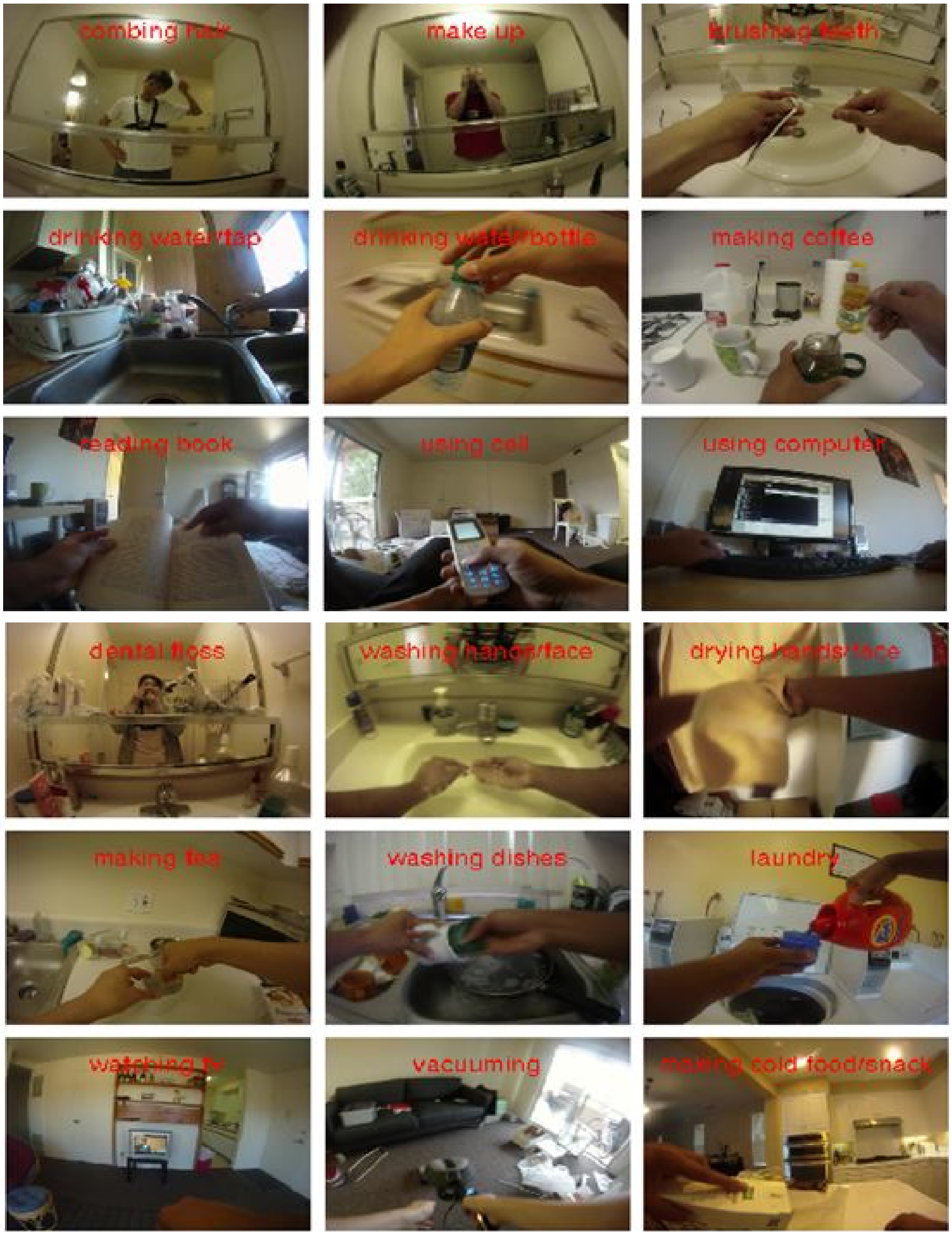


Figure 4.2: Overview of the 18 activities annotated in the ADL dataset

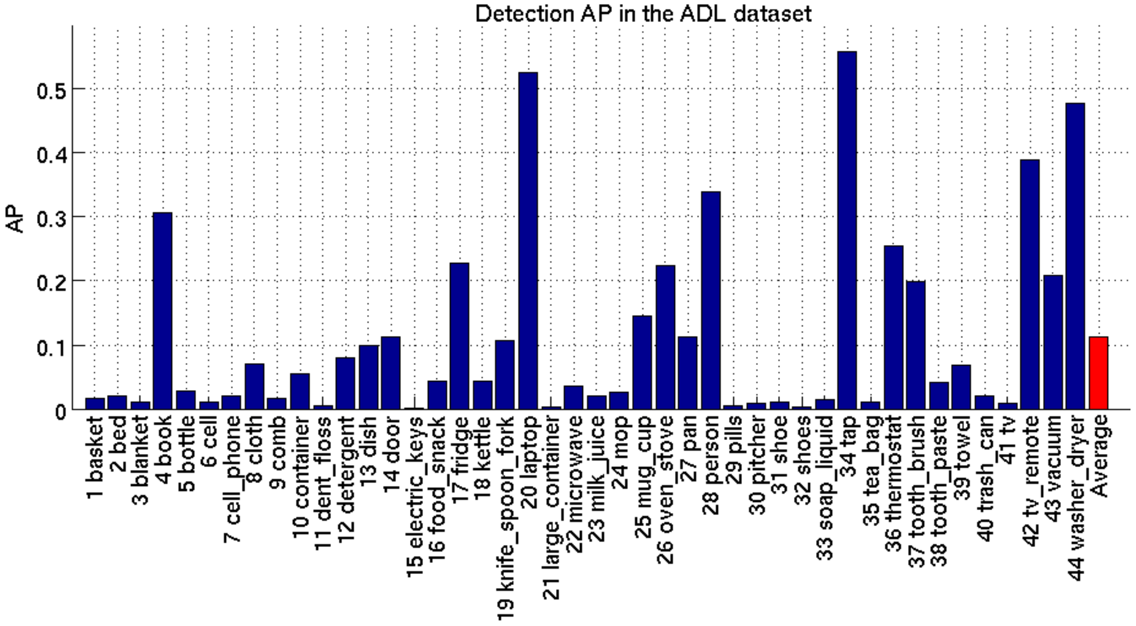


Figure 4.3: Results of object recognition

4.2.3 Place Recognition results

In this section, we report the results obtained on the ADL dataset for the place recognition module. We use a Chi-Square kernel and retain 500 dimensions for the KPCA. We compared two different types of annotation of the environment: a room based annotation compound of 5 classes (bathroom, bedroom, kitchen, living room, outside) and a place based annotation compound of 7 classes (in front of the bathroom sink, in front of the washing machine, in front of the kitchen sink, in front of the television, in front of the stove, in front of the fridge and outside).

We have obtained average accuracies of 58.6% and 68.4%, for the room and place recognition, respectively.

Hence, we can conclude that, for this dataset, place recognition is more suitable than room recognition. We believe this is due to the fact that the concept of place has a smaller variability than the concept of room in the appearance space. Hence, in the following, we will use place recognition in our experiments.

4.2.4 Activity Recognition results

In this section we show our results in daily activity recognition in egocentric videos. As already mentioned, our system identifies the activity at every frame of the video using a sliding window, what allows us to compute Average Frame level classification accuracy. To this end, we have also included a new class “no activity” associated to frames that are not showing any activity of interest. It is also worth noting that the global performance is computed by averaging the particular accuracies for each class (rather than simply counting the number of correct decisions) and, thus, adapts better to highly unbalanced sets as the one being used (where most of the time there is no activity of interest).

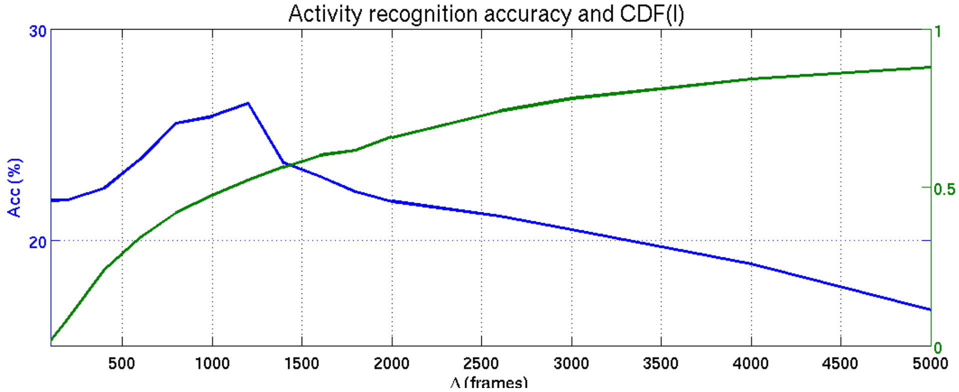


Figure 4.4: Activity Recognition Accuracy with respect to the window size Δ used in the analysis

Approach	Avg Fr. Acc	AvgSeg. Acc
Our Active Objects	24,2%	40,5%
Our Places	19,7%	11,1%
Our Active Objects + Places	26,3%	41,3%
Pirsiavash et al. [Pirsiavash12]	23%	36,9%

Table 4.1: Activity recognition accuracy for our approach computed at Frame and Segment level, respectively.

In our first experiment, we have studied the influence of the window size Δ. Based on the results shown in Figure 4.4 (blue line), we can draw interesting conclusions: on the one hand, too short windows do not model the dynamics of an activity, understood in our case as sequences of different active objects or places. Oppositely, too long windows may contain video segments showing various activities. Although, from our point of view, this fact might help to detect several strongly related activities by reinforcing the knowledge about one activity by the presence of the other (e.g. washing hands/face and drying hands/hair are activities that usually occur following the same temporal sequence), it might also lead to features containing too many active objects and places. These features would therefore make these frames difficult to assign to a particular activity. In our case, the value that best fits the activities in ADL dataset is $\Delta = 1200$ frames, which corresponds to approximately 47 seconds of video footage. In fact, looking at the cumulative distribution of the activities length in the dataset (green line in Figure 4.4), we have found a median value of approximately 1100 frames, so that the best results in activity recognition are achieved for Δ values around this point.

Secondly, we have also assessed the importance of each feature for recognizing activities. In the first column of Table 4.1, we show the results of our approach using either just active object or place detectors, and using an early combination of both of them by feature concatenation. As one can notice from the results, combining objects and their context (the place where they are located) notably improves the performance achieved by simply using the object detectors. Let us note that we have also tested a late fusion scheme that did not lead to improvements in the system performance.

Furthermore, for comparison, we also include the results obtained with the software provided by the authors of [4.1]. This approach uses the outputs of various detectors of active and non-active objects implemented using the Deformable Part Models (DPM) [4.2]. Let us note that, as mentioned by the authors in the software, results differ from the ones reported [4.1] due to changes in the dataset. From the results, and due to the similar classification pipeline of both methods, we can conclude that our features are more suitable for the activity recognition problem.

Finally, we additionally include results of a segment based evaluation in which ground truth time segmentations of the video are available in both training and testing steps. Hence, this case simplifies the activity recognition from a category segmentation problem to a simple classification problem for each segment. Following the evaluation protocol in [4.1], this case lacks the 'no activity' class, so that only video intervals showing activities of interest are taken into account. Combining objects and context provides the best performance, which is again superior to the one obtained by [4.1].

These results lead us to conclude that, recognizing activities in egocentric video does not require identifying every object in a scene, but simply detect the presence of "active" objects and provide a compact representation of the object context. This context has been implemented in this work by means of a global classifier of the place.

4.3 Conclusions and Further Work

In this section we have shown how activity recognition in egocentric video can be successfully addressed by the combination of two sources of information: a) active objects either manipulated or observed by the user provide very strong cues about the action, and b) context also contributes with complementary information to the active objects, by identifying the place in which the action is being made.

To that end, an activity recognition method that models activities as sequences of active objects and places have been used on a challenging egocentric video dataset showing daily living scenarios for various users. Under two different scenarios, we have demonstrated how the combination of both objects and context provides notable improvements in the performance and that it outperforms state-of-the-art methods using active and passive objects representations.

References

[4.1] H. Pirsiavash et D. Ramanan, «Detecting Activities of Daily Living in First-person Camera Views,» chez IEEE Conference on Computer Vision and Pattern Recognition, 2012.

[4.2] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester et D. Ramanan, «Object Detection with Discriminatively Trained Part-Based Models,» IEEE Transactions on Pattern Analyisis and Machine Intelligence, vol. 32, n° %19, pp. 1627-1645, 2010.

[4.3] H. Boujut, J. Benois-Pineau et R. Megret, «Fusion of multiple visual cues for visual saliency extraction from wearable camera settings with strong motion,» chez ECCV 2012 - Workshops, 2012.

[4.4] G. Csurka, C. R. Dance, L. Fan, J. Willamowski et C. Bray, «Visual categorization with bags of keypoints,» chez In Workshop on Statistical Learning in Computer Vision, ECCV, 2004.

[4.5] H. Bay, A. Ess, T. Tuytelaars et L. {Van Gool}, «Speeded-Up Robust Features (SURF),» Comput. Vis. Image Underst., vol. 110, pp. 346-359, June 2008.

[4.6] C. Cortes et V. Vapnik, «Support-vector networks,» Machine Learning, vol. 20, pp. 273-297, 1995.

[4.7] V. Sreekanth, A. Vedaldi, C. V. Jawahar et A. Zisserman, «Generalized {RBF} feature maps for efficient detection,» chez Proceedings of the British Machine Vision Conference (BMVC), 2010.

[4.8] J. C. Platt, «Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods,» chez Advances in Large Margin Classifiers, 1999.

[4.9] A. Pronobis, O. M. Mozos, B. Caputo et P. Jensfelt, «Multi-modal Semantic Place Classification,» The International Journal of Robotics Research (IJRR), Special Issue on Robotic Vision, vol. 29, n° %12-3, pp. 298-320, #feb# 2010.

[4.10] V. Dovgalecs, R. Megret et Y. Berthoumieu, «Multiple Feature Fusion Based on Co-Training Approach and Time Regularization for Place Classification in Wearable Video,» Advances in Multimedia, 2013.

[4.11] B. Scholkopf, A. Smola et K.-R. Muller, Nonlinear component analysis as a kernel eigenvalue problem, 1996.

5 Ontology Patterns for Behaviour Modelling

Automated activity recognition has been a major challenge for context-awareness. In many application domains though, affording meaningful user-tailored feedback requires more than the identification of the user's ongoing activity. This is particularly relevant in healthcare and assisted living applications, where promoting the user's well-being necessitates a broader understanding of user behaviour, including not only what activities are performed, but also idiosyncratic and habitual knowledge such as the manner in which an activity is performed and recurrent patterns of activities (e.g. bed time routine).

As described in Section 2.3 several ontology-based approaches have been proposed for modelling behavioural aspects aiming to avail of the well-defined semantics and automated inference services. However, staying at the level of instances, these efforts fail to provide reusable formal models of behaviour, thus hindering interoperability.

In the following, we present ontology-based patterns for modelling various behavioural aspects [5.1] capturing either already known information (e.g. average breakfast duration as reported by the PwD or informal carer) or dynamically learned one (e.g. average breakfast duration based on Dem@Care monitoring and interpretation). The proposed patterns implement the descriptions and situations (DnS) ontology pattern [2.3.32] of DOLCE Ultra Lite[2] (DUL) and make use of the meta-modelling capabilities of OWL 2, namely *punning*. The latter enables to treat domain activity concepts as instances and hence allows for property assertions to be made among activity types. In that way, the proposed behaviour patterns enable the representation of contextualised views and afford reusable pieces of knowledge that cannot otherwise be directly expressed by the standard ontology semantics. DOLCE provides a formal modelling basis and has been used for a number of core ontologies such as [5.2][5.3][5.4], while the pattern-oriented approach of DUL provides native support for modularisation and extension by domain specific ontologies.

Currently, six behavioural patterns have been implemented, three for formalising structural relations that allow to express the various manners in which a PwD may carry out an activity and three for formalising habitual knowledge, such as frequency and duration of activities.

The rest of the Section is structured as follows. Section 5.1 presents the functional and non-functional requirements on the ontology-patterns behaviour model. Section 5.2 introduces the core activity pattern that extends the DnS pattern that serves as the conceptual base for the definition of behavioural patterns, while Sections 5.3 to 5.8 describe the individual behaviour patterns and demonstrate their use.

5.1 Behaviour patterns requirements

To design and engineer the behavioural patterns, functional and non-functional requirements were derived and analysed. These were based on the study of existing ontology-based approaches for capturing meta-knowledge and ontology design patterns[3], related work on

[2] http://www.loa.istc.cnr.it/ontologies/DUL.owl

[3] http://ontologydesignpatterns.org/

ontology-based approaches to the modelling of behaviour aspects, as described in Section 2.3 and of course on the use case scenarios targeted by Dem@Care.

5.1.1 Functional requirements

A key feature of the Dem@Care project is to enable the customised interpretation of PwD behaviour and the delivery of appropriately tailored management and support services. To accomplish this, the system needs to be aware not only of the activities a PwD is carrying out, but also of diversions from its usual behaviour as well as of clinically relevant incoherencies. Typical examples of such incoherencies include among others errors related to organisation, realisation, sequence and completion aspects [5.4].

Organisation errors happen when the patient performs some steps of an activity in an inappropriate way. For instance, the patient can use the wrong type of spoon, or even a knife, to mix up the ingredients of a receipt. Realisation errors happen when, due to a distraction or a memory lapse, the person performs actions that are unrelated with their ongoing activity or original goal, or skips some steps of his activity. For example, a patient can put a bowl of soup in the microwave oven in order to heat it while forgetting to start the microwave and, a few minutes later, eat the soup thinking that it is hot. Sequence errors correspond to some disorganization in the course of the activity's steps. For instance, the patient can try to change the television channel without having turned it on beforehand. Completion errors happen when the patient is unable to finish the ongoing task, because they stop half ways through it or because they indefinitely repeat one or more steps of the task. For instance, a patient may be in the process of making tea and opens a kitchen cupboard in order to take a cup, but, instead, may begin to repetitively open and close the cupboard.

To support the recognition of situations like the aforementioned ones, a behavioural model should account for a number of aspects pertinent to the various manners in which a PwD may carry out activities of daily living, including:

- the steps (sub-activities) that comprise an activity (e.g. Alice has her tea either plain or with milk),
- temporal/spatial patterns such as sequence patterns (e.g. afternoon rest for Paul consists in serving himself a glass of wine, turning on the CD player, and sitting on his armchair),
- initiation and termination patterns (e.g. Lauren starts her breakfast by turning on the kettle and completes it by putting the dishes in the dishwasher),
- information about the typical frequency of a behavioural element (e.g. Lauren has three cups of tea daily),
- information about the typical duration of a behavioural element (e.g. Alice's dinner takes on average an hour),
- information about how many times an activity is repeated within a certain context (e.g. how many times the patient opens and closes the CD player when they try to play music).

5.1.2 Non-functional requirements

In addition to the afore-described functional requirements, a number of non-functional requirements are derived from reported and own experience.

- *Extensibility.* As new developments and functional requirements emerge, the behaviour patterns model should be able to include additional aspects.
- *Axiomatisation.* To establish a common understanding and ensure interoperability through machine accessible semantics, the proposed ontology-based behaviour patterns need to be sufficiently formal so that systems can reason about the represented knowledge and carry out semantic checks on its validity.
- *Modularity.* While the behaviour model needs to capture different aspects of structural knowledge, applications will commonly use only portions of it. A modular design allows for selecting the parts of the model used.
- *Reusability.* The behaviour patterns model shall be able to incorporate existing domain ontologies and make use of that domain knowledge rather than requiring remodelling it, while supporting reuse of its modules despite the different viewpoints imposed by different domains.
- *Separation of concerns.* A core model needs to be applicable for arbitrary application domains and enable integration and reuse of models of domain-specific knowledge. To this end, the domain-independent knowledge in the core model needs to be clearly separated from the domain-specific knowledge.

In the following, we describe the proposed ontology-based behaviour patterns and illustrate their use. With respect to the aforementioned functional requirements, the composition pattern and the specialisation pattern implement the structural relationships between activities, allowing in addition for modelling of spatiotemporal information. The boundary pattern allows for modelling information regarding the activities that initiate and terminate a given behaviour. The frequency pattern and the duration pattern provide for modelling habitual frequency and duration information associated with an activity. The repetition pattern specialises further the frequency pattern and allows for modelling frequency information with respect to specific context (e.g. location).

5.2 Core Activity Pattern

The DnS design pattern provides a principled approach to context reification through a clear separation of states-of-affairs, i.e. a set of assertions and their interpretation based on a non-physical context, called a *description* [2.3.32]. Intuitively, DnS axioms try to capture the notion of situation as a unitarian entity out of a state of affairs, with the unity criterion being provided by a description. In that way, when a description is applied to a state of affairs, a situation emerges.

In the proposed ontology-based behaviour patterns, we use DnS to formally provide precise representations of contextualized situations and descriptions on activity concepts of domain ontologies, describing the different activity types and contextual relations that can be associated with complex domain activities. The modelling capabilities have been designed with a minimum of semantic commitment to guarantee maximal interoperability. As such, the behaviour patterns can reuse existing foundational ontologies for modelling different aspects of activities, e.g. entities, places, such as SEM [5.5] and Ontonym [5.6].

The implementation of DnS in DUL allows the relation of situations and descriptions with individuals of the dul:Event and dul:EventType classes, respectively. For example, the Event-Model-F [5.7] implements a number of instantiations on top of the DnS pattern to describe relations among asserted events (instances of the dul:Event class), such as causality and correlation. In contrast, the scope of the core activity pattern is to conceptually describe the activity context that defines complex activities at the class level, and not to represent relations directly among activity instances.

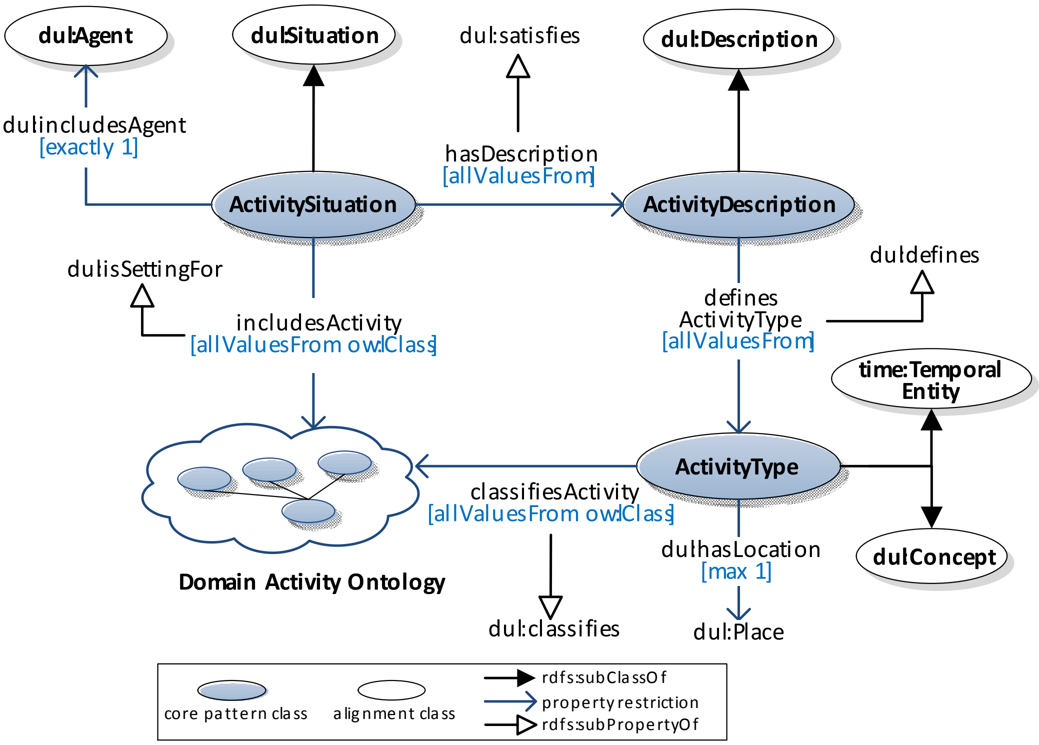


Figure 5.1: Core Activity Pattern

To this end, the core activity pattern allows the representation of the following activity-related conceptualisations, as illustrated in Figure 5.1, where classes belonging to the defined behaviour patterns are highlighted in the blue to show the alignment with classes of DUL.

- **Activity situations.** An activity situation defines a set of activity classes that are involved in a specific pattern instantiation (includesActivity property) and they are interpreted on the basis of an activity description (hasDescription property). Each situation is also correlated with one user/agent (dul:includesAgent property).
- **Activity descriptions.** An activity description serves as the descriptive context of an activity situation, defining the activity types (definesActivityType property) that classify the domain activities of a specific pattern instantiation, creating views on situations.
- **Activity types.** Activity types are DUL concepts that classify activity classes, i.e. they treat domain activity classes as instances, describing how they should be interpreted in a particular situation. These descriptions mainly involve the specification of the temporal constraints that characterise the respective contextual activities, reusing the temporal property assertions provided by the OWL-Time ontology in terms of the

time:TemporalEntity[4] class. The dul:hasLocation property of DUL is used to correlate an activity type with a location (dul:Place).

Roughly speaking, the definition of a behaviour pattern is specified in two levels of granularity: (a) the *situation*, that provides an abstract description of the behaviour in terms of the domain activity types that are involved, and (b) the *description*, that can be thought as a descriptive context that classifies the activity classes of the situation in order to create a view, i.e. to define the contextual relations that characterise a specific behaviour.

5.3 Activity Composition Pattern

The activity composition pattern enables to formally express behavioural information related to the way a PwD carries out complex activities, namely activities that involve two or more steps and are defined as the composition of atomic or other complex activities, such as making tea or preparing pasta.

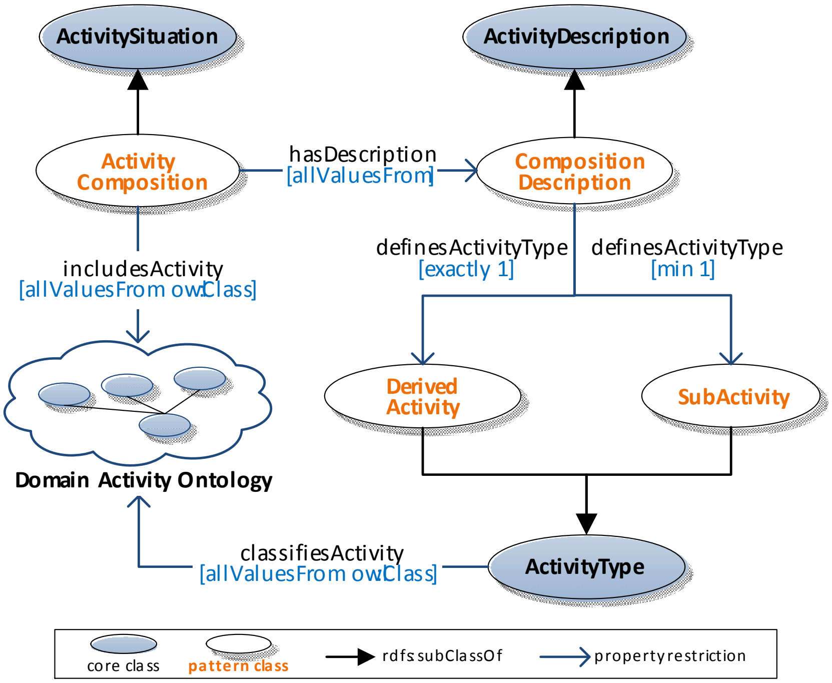


Figure 5.2: The Activity Composition pattern

As shown in Figure 5.2, a composite activity definition is expressed by an ActivityComposition that satisfies a CompositionDescription. The situation includes the descriptive context that admits the composition, namely the composite activity, its constituent activities and their pertinent spatiotemporal correlations. The classes DerivedActivity and SubActivity express the complex activity to be inferred and its constituent activities, respectively. Temporal correlations among the involved activities are expressed through their

[4] http://www.w3.org/2006/time

associated DerivedActivity and SubActivity classifications that subsume the ActivityType class, and thus the time:TemporalEntity concept (see Figure 5.1).

Using the proposed pattern, one can express, for example, that making tea amounts for Alice to boiling water, place tea bag in cup and adding sugar in that sequence in kitchen (see Figure 5.3), while for Paul making tea includes also the addition of milk.

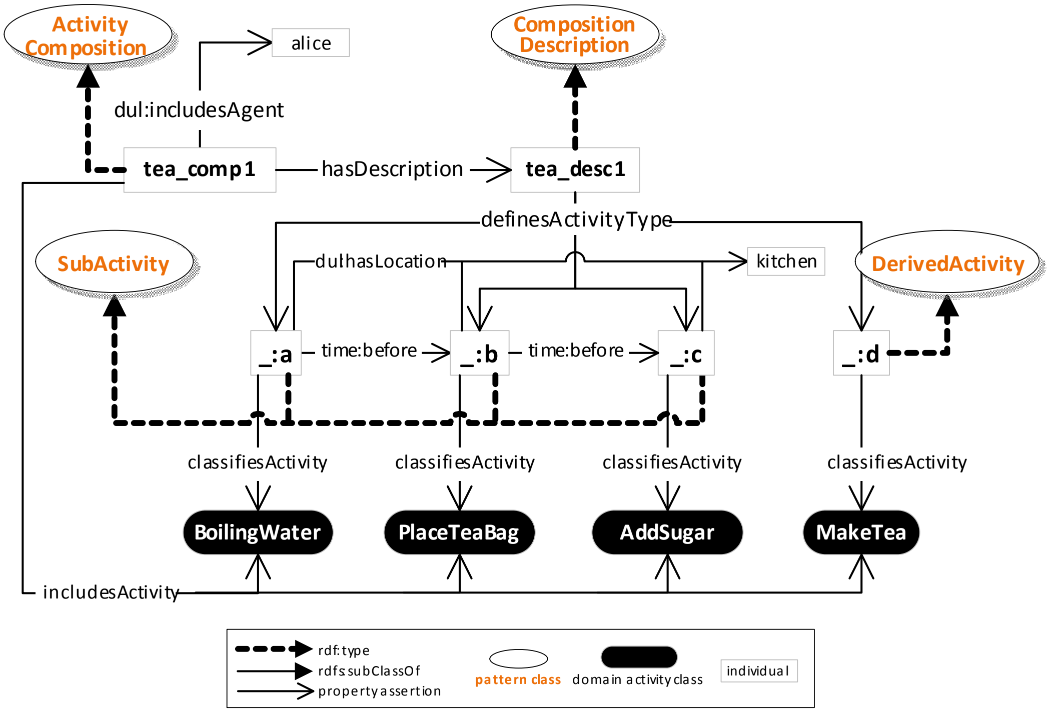


Figure 5.3: Example instantiation of the composition pattern

5.4 Activity Specialisation Pattern

The activity specialisation pattern enables to formally express behavioural information related to the way a PwD carries out complex activities, but unlike the activity composition pattern, the specialisation pattern refers to complex activities that are derived as further specialisations of a given atomic or other complex activity.

As shown in Figure 5.4 a definition of this type is expressed by an ActivitySpecialisation that satisfies a SpecialisationDescription. The situation includes the descriptive context that admits the specialisation, namely the activity that is subject to further specialisation, one or more associated activities, and their pertinent temporal correlations. The classes SpecialisedType and SpecialisationType express the asserted and derived activity respectively, while the class SpecialisationContext allows expressing activities comprising the descriptive context. Temporal correlations among activities are expressed through their associated SpecialisedActivity, SpecialisationType and ContextSpecialisation classifications that subsume the ActivityType class, and thus the time:TemporalEntity concept.

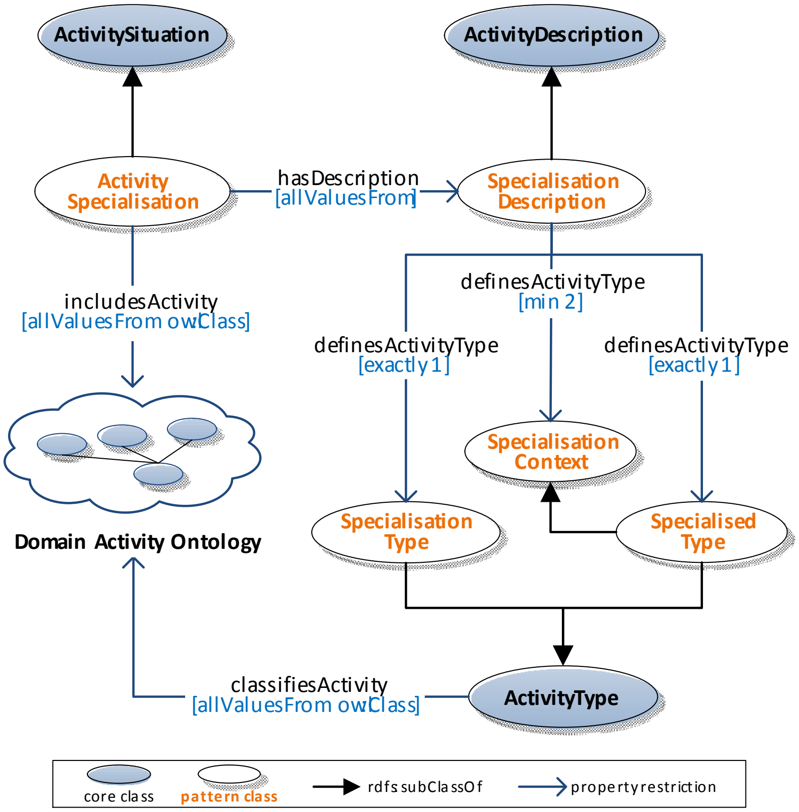


Figure 5.4: Activity Specialisation Pattern

Figure 5.5 illustrates an example instantiation of the specialisation pattern for NightBathroomVisit. Intuitively, the instantiation of the pattern defines that an InBathroom instance is further specialised as a NightBathroomVisit, if it is temporally contained in a NightSleep activity (the dul:includesAgent and dul:hasLocation properties are omitted for simplicity).

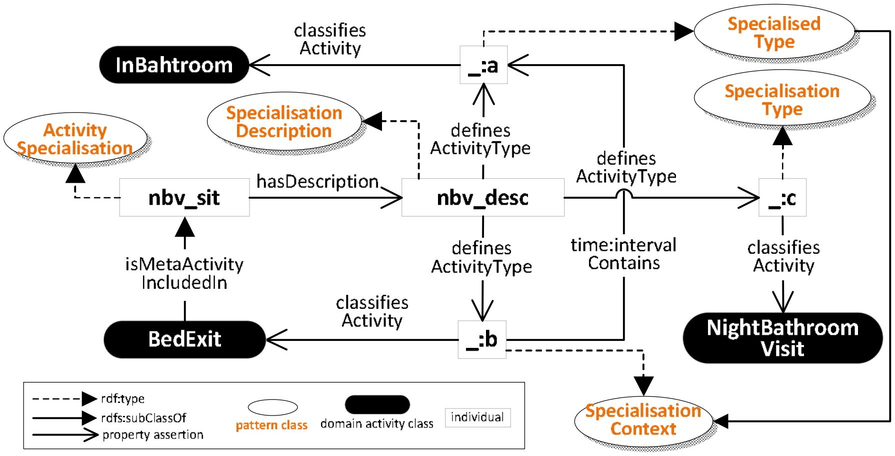


Figure 5.5: Example instantiation of the Activity Specialisation pattern

5.5 Activity Boundaries Pattern

The activity boundaries pattern, shown in Figure 5.6, enables to formally express behavioural information related to the way a PwD carries out complex activities, with respect to the initiating and terminating activities that characterise a complex activity.

Formally, an ActivityBoundaries situation includes one instance of an activity that is classified by the concept BoundedActivity, one instance of an activity that is classified as InitialActivity and one instance of an activity that is classified as TerminalActivity. Accordingly, an ActivityBoundaries situation satisfies a BoundaryDescription that defines the concepts BoundedActivity, IntialActivity and TerminalActivity for classifying the reference activities, i.e. the activity for which start and end information is provided, and its respective initiating and terminating steps, respectively.

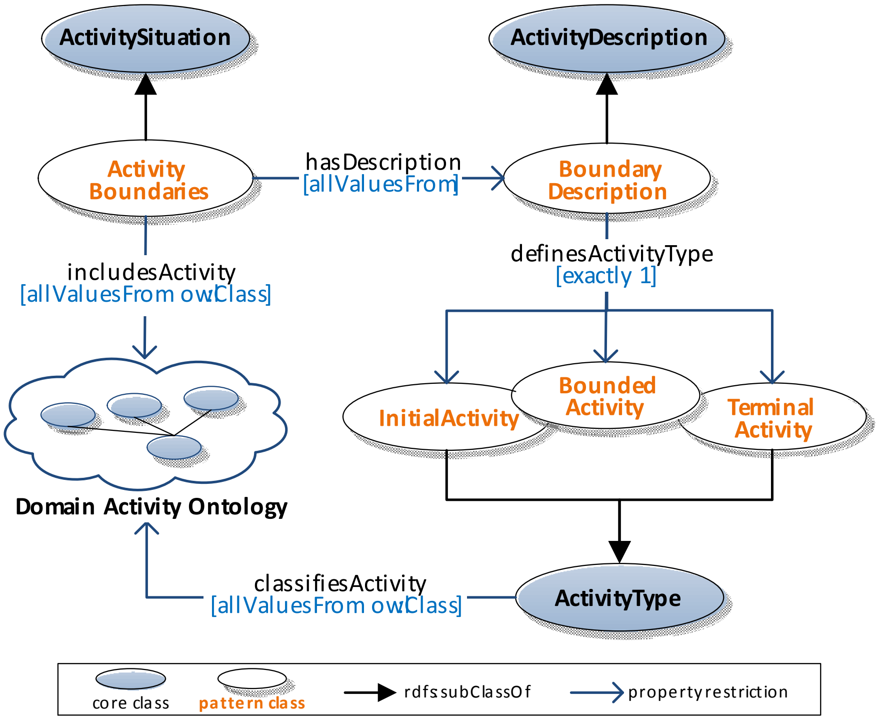


Figure 5.6: The Activity Boundaries pattern

Figure 5.7 illustrates an example instantiation of the boundaries pattern, assuming that for a given PwD making breakfast starts with turning on the kettle and completes with putting the dishes in the dishwasher (the dul:includesAgent and dul:hasLocation properties are omitted from the figure for simplicity). If there are more than one initiation-termination patterns, they can be captured through additional BoundaryDescription instances.

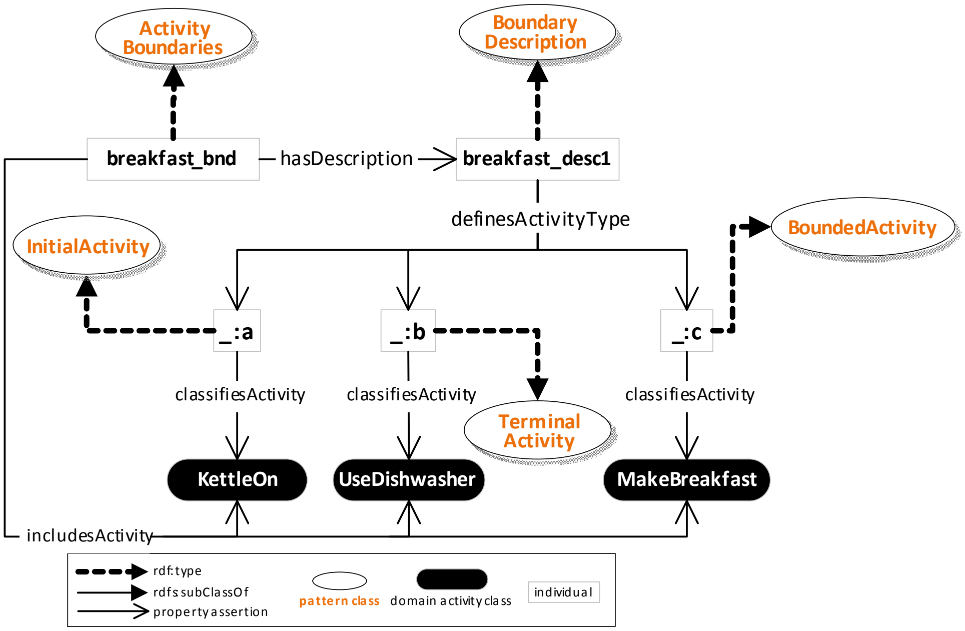


Figure 5.7: Example instantiation of the Activity Boundaries pattern

5.6 Activity Repetition Pattern

The activity repetition pattern allows expressing the number of repetitions of certain activities, within the context of a complex activity (e.g. how many times the PwD interacts with given objects during the preparation of a meal).

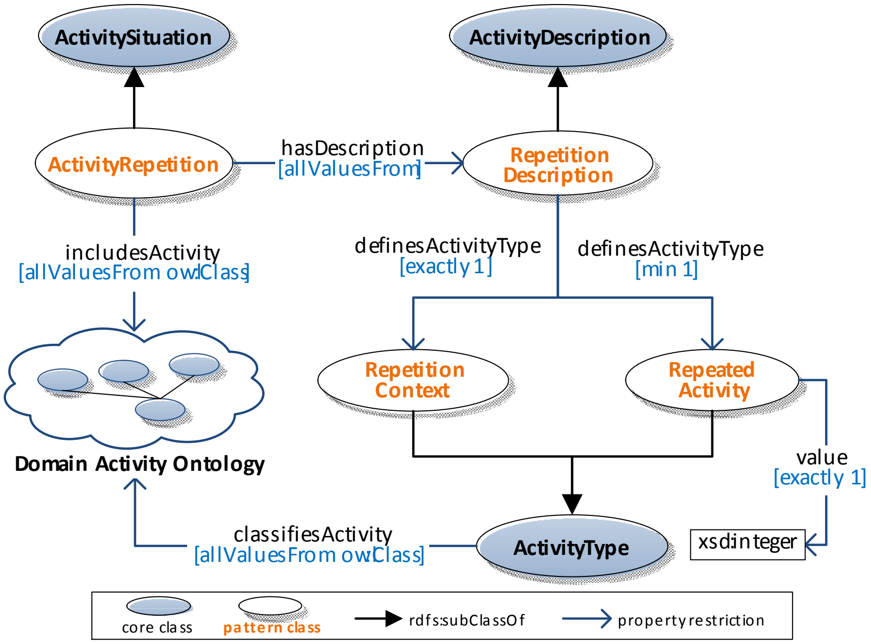


Figure 5.8: The Activity Repetition pattern

As shown in Figure 5.8, an ActivityRepetition situation includes one instance of an activity that is classified as a RepetitionContext concept, namely the complex activity context within which we examine the repetition of activities, and one or more instances that are classified as RepeatedActivity and correspond to the activities that are being repeatedly performed. The number of times that an instance of the RepeatedActivity is performed is captured through the datatype property value. Accordingly, an ActivityRepetition situation satisfies a RepetitionDescription that defines the concepts RepetitionContext and RepeatedActivity.

The motivation behind this pattern is to allow for modelling already known behavioural incoherencies in order to monitor their evolution in the course of time. Figure 5.9 illustrates an example instantiation of the repetition pattern, capturing a situation where during breakfast preparation, the kettle is turned on 5 times (the dul:includesAgent and dul:hasLocation properties are omitted from the figure for simplicity).

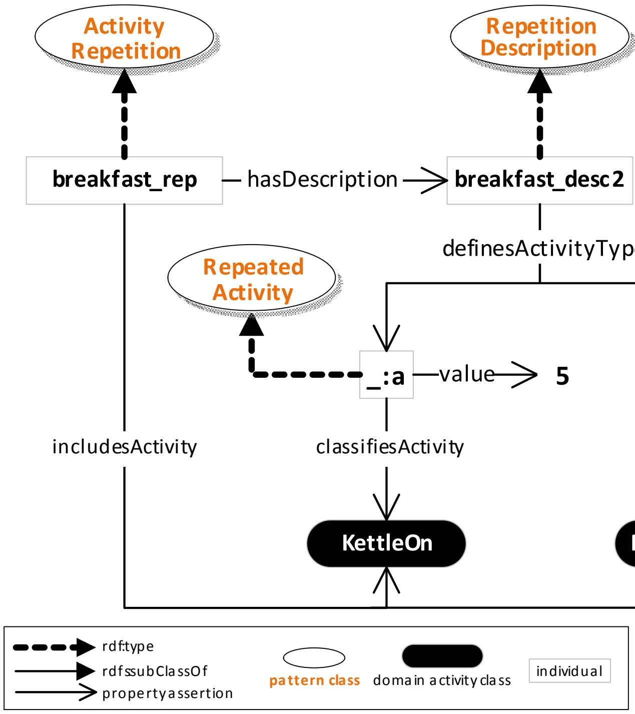


Figure 5.9: Example instantiation of the Activity Repetition pattern

5.7 Activity Frequency Pattern

The activity frequency pattern allows for modelling the frequency of an activity. For example, it can be used to describe how many times the PwD visits the bathroom during a day or how many times she goes out in a month.

As shown in Figure 5.10, an ActivityFrequency situation includes one instance of an activity that is classified as a FrequencyContext concept, namely the activity whose frequency is described, two property assertions, period and value, that express the timescale (daily, weekly, monthly) and the frequency, respectively. Accordingly, an ActivityFrequency situation satisfies a FrequencyDescription that defines the concept FrequencyContext.

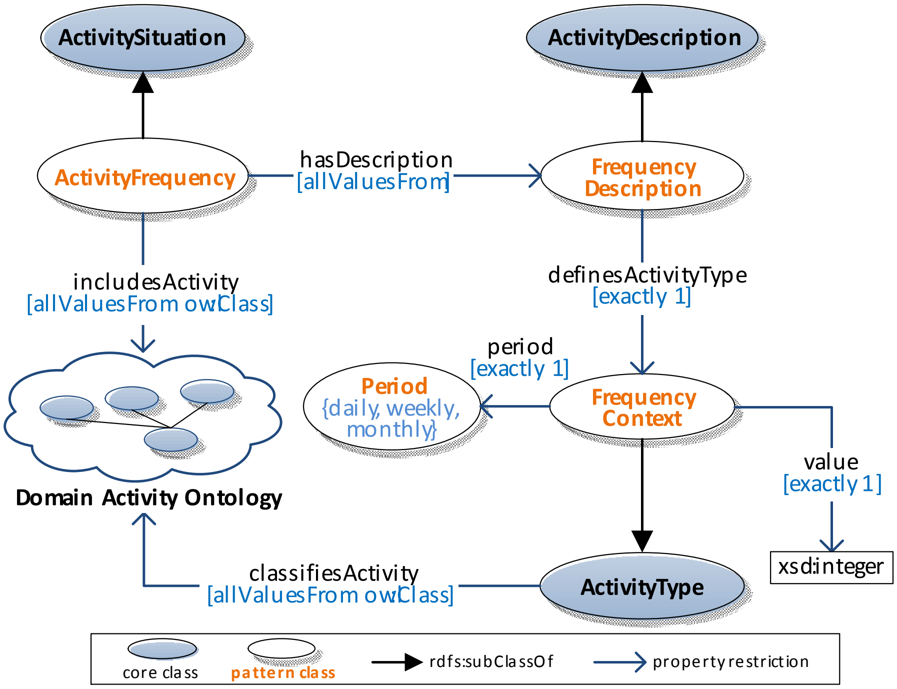


Figure 5.10: Activity Frequency Pattern

An example instantiation of the pattern illustrating the frequency of meals per day is shown in Figure 5.11 (the dul:includesAgent and dul:hasLocation properties are omitted from the figure for simplicity).

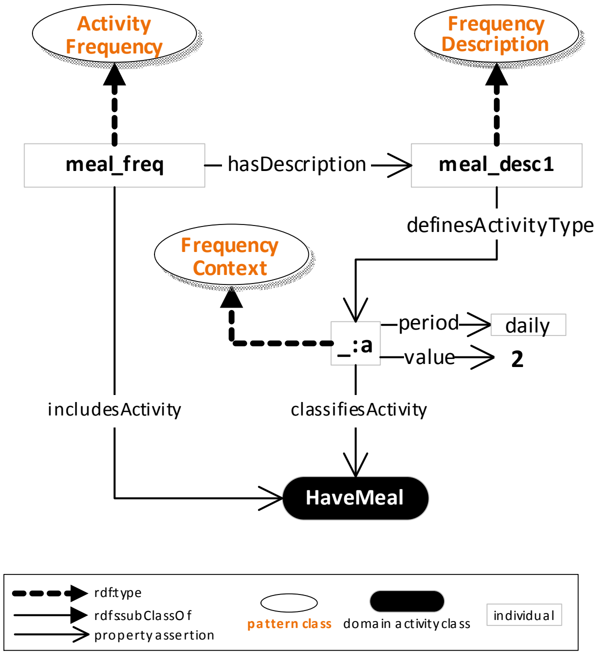


Figure 5.11: Example instantiation of the Activity Frequency pattern

5.8 Activity Duration Pattern

The activity duration pattern allows for the association of descriptive contexts to domain activities relevant to the duration. For example, it can be used to describe how long it usually takes the PwD to finish a meal or to take a bath.

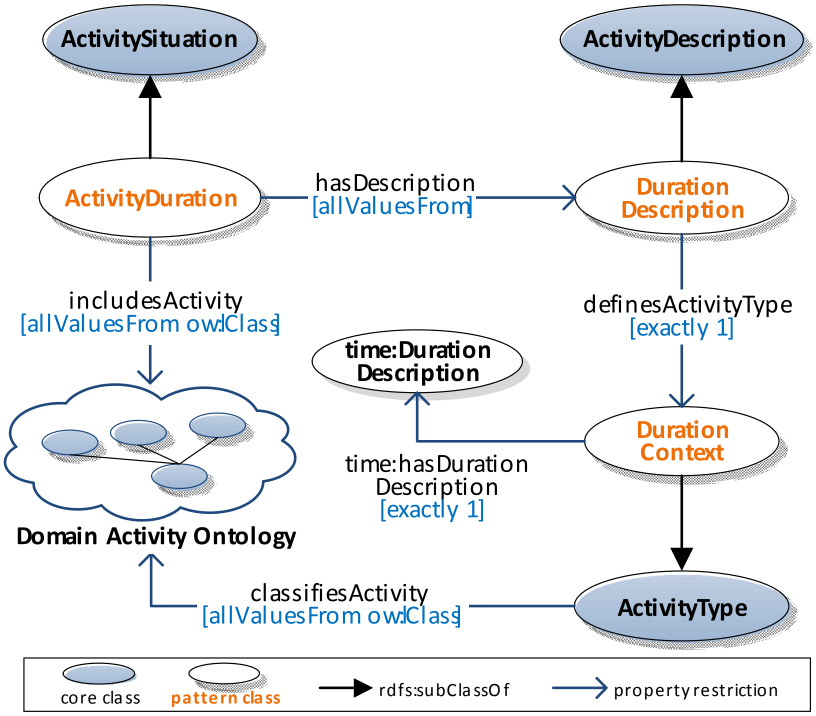


Figure 5.12: The Activity Duration pattern

As shown in Figure 5.12, an ActivityDuration situation includes one instance of an activity that is classified as a DurationContext concept, namely the activity whose duration is described and a property assertion, namely time:hasDurationDescription, to express the actual duration using the OWL Time vocabulary. Accordingly, an ActivityDuration situation satisfies a DurationDescription that defines the concept DurationContext.

An example instantiation of the pattern illustrating the normal duration of a meal is shown in Figure 5.13 (the dul:includesAgent and dul:hasLocation properties are omitted from the figure for simplicity).

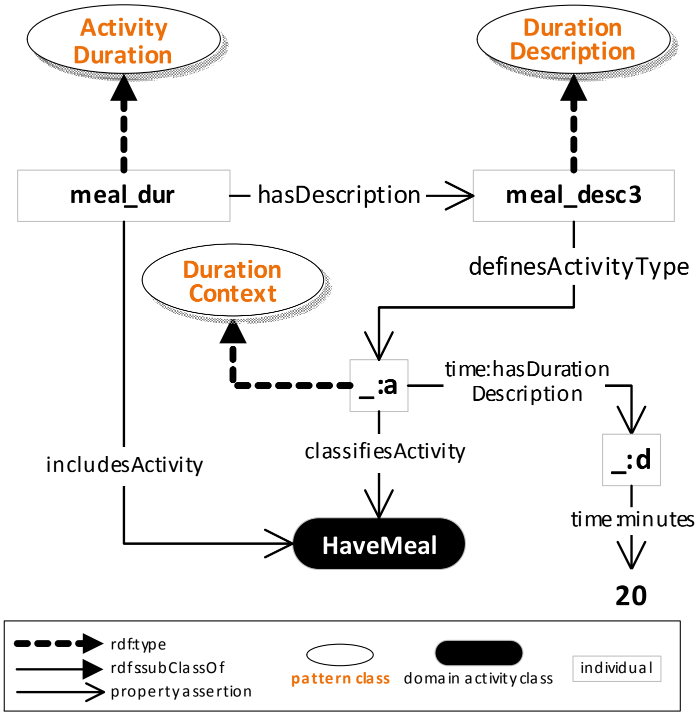


Figure 5.13: Example instantiation of the Activity Duration pattern

5.9 Discussion and Future Work

In this Section we presented an ontology-based pattern-oriented approach for the formal modelling of user behavioural aspects. We defined a core activity pattern that extends DnS and provides the conceptual basis for structuring the individual behaviour pattern modules and six specialised patterns that allow for modelling the manner in which activities are carried out (e.g. sequence information, start/end activity patterns) as well as habitual information in terms of frequency, repetition and duration information.

The adopted DnS pattern-oriented implementation provides native support for modularisation and extension by domain specific ontologies (e.g. domain activity ontologies, routine ontologies). Moreover, alignment with the foundational ontology DUL provides a further basis for future extensions; in addition, adopting the formal semantics of the foundational ontology, the more specific semantics of the concepts and relations defined in the proposed pattern-based behaviour model can be validated.

In addition to serving as formal, reusable models for capturing user behaviour, the defined patterns can be also utilised for behaviour recognition purposes, as the patterns capture the contextual conditions and the spatiotemporal relations that characterise complex activities. The adopted DnS and DUL compliant design principles, enable to reuse the encapsulated semantics across applications with similar scope but different implementation frameworks, by translating the pattern-based models into the respective framework language.

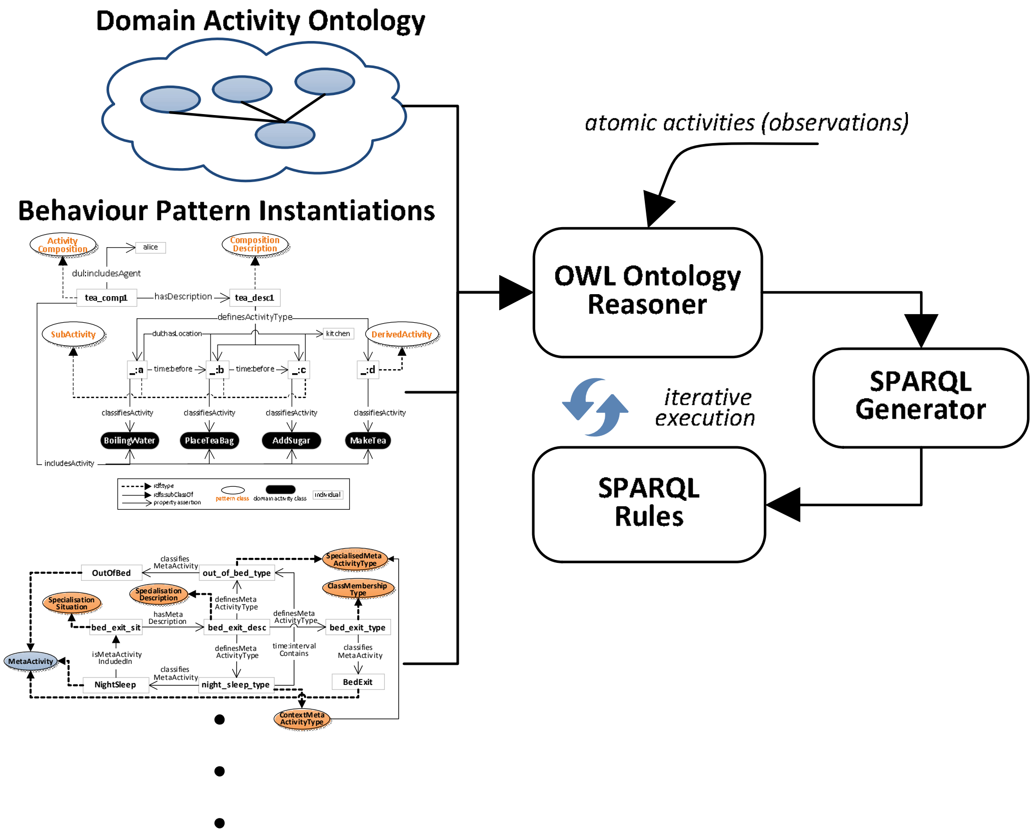


Figure 5.14: SPARQL generation and execution framework

In Dem@Care for instance, a SPARQL-based implementation is followed by the Semantic Interpretation (SI) component [5.8]. As depicted in Figure 5.14 that illustrates the abstract architecture of the recognition framework, the semantics of the Dem@Care Event/Entity ontology [5.9] for representing domain events/activities and the semantics of the instantiated patterns, e.g. the property restrictions, sub-properties, inverse properties, etc. are first handled by an ontology reasoner, which in the SI implementation corresponds to the OWLIM ontology reasoner [5.10]. The ontology model is then used by a SPARQL Generator to dynamically generate SPARQL rules, based on the provided pattern instantiations.

In the case of the specialisation pattern for example, the activity classes that are classified by SpecialisationContext are used to define the triple patterns that match the corresponding activity instances in the WHERE clause. Additionally, the SpecialisationType that classifies the class of the specialisation is used to define the triple patterns in the CONSTRUCT clause that specify the additional class type of the specialised instance. Figure 5.15 shows the SPARQL rule that is generated for the recognition of specialisation pattern instantiation example of night bathroom visit that was presented in Section 5.4.

```
CONSTRUCT{
    ?ib a :NightBathroomVisit
}
WHERE {
    ?be a :BedExit ;
            :hasAgent ?pwd ;
            :startTime ?be_start ;
            :endTime ?be_end .
      ?ib a :InBathroom;
            :hasAgent ?pwd ;
      :startTime ?ib_start .
    FILTER (:contains(?be_start, ?be_end, ?ib_start)) .
    FILTER NOT EXISTS {?ib a :NightBathroomVisit . } .
}
```

Figure 5.15: SPARQL rule for deriving NightBathroomVisit instances

As noted in D5.2 [5.8] however, in the first version of SI the focus had been on the aggregation and semantic correlation of the descriptions extracted from the analysis components of the Dem@Care system, assuming that perfect information is available and without taking into account uncertain, missing information and conflicts. Currently, the use of patterns is investigated under more flexible reasoning schemes that will allow taking into account for the uncertainty present in the input observations, including temporal incoherencies, missing information, and erroneous analysis results. First results will be reported in the upcoming deliverable D5.4.

References

[5.1] G. Meditskos, S. Dasiopoulou, V. Efstathiou, I. Kompatsiaris, *"Ontology Patterns for Complex Activity Modelling"*, 7th International Web Rule Symposium: Research Based and Industry Focus (RuleML), Seattle Metropolitan Area, USA, July 11-13, 2013

[5.2] D. Oberle, A. Ankolekar, P. Hitzler, P. Cimiano, M. Sintek, M. Kiesel, B. Mougouie, S. Baumann, S. Vembu, M. Romanelli, *"DOLCE ergo SUMO: On foundational and domain models in the SmartWeb Integrated Ontology (SWIntO)"*, J. Web Sem. 5(3): 156-174, 2007

[5.3] R. Arndt, R. Troncy, S. Staab, L. Hardman, M. Vacura, *"COMM: Designing a Well-Founded Multimedia Ontology for the Web"*, ISWC/ASWC, p. 30-43, 2007

[5.4] C. Baum, D. Edwards, *"Cognitive performance in senile dementia of the Alzheimer's type: The kitchen task assessment"*, The American Journal of Occupational Therapy, Vol. 47 (5), 1993

[5.5] W. R van Hage, V. Malaise, R. Segers, L. Hollink, G. Schreiber, "*Design and use of the Simple Event Model (SEM)*", J. Web Sem. 9(2), p. 128-136, 2011

[5.6] G. Stevenson, S. Knox, S. Dobson, P. Nixon, "*Ontonym: a collection of upper ontologies for developing pervasive systems*", 1st Workshop on Context, Information and Ontologies, p. 9:1-9:8, 2009

[5.7] A. Scherp, T. Franz, C. Saathoff, S. Staab, *"F - A model of events based on the foundational ontology dolce+DnSultralight"*, K-CAP, p. 137-144, 2009

[5.8] S. Dasiopoulou, V. Efstathiou, G. Meditskos, C. Crispim, A.T. N., V. Buso, "D5.2 Multi-parametric Behaviour Interpretation v1", Dementia Ambient Care: Multi-Sensing Monitoring for Intelligent Remote Management and Decision Support, Dem@Care – FP7 288199.

[5.9] S. Dasiopoulou, V. Efstathiou, G. Meditskos, *"D5.1 Semantic Knowledge Structures and Representation"*, Dementia Ambient Care: Multi-Sensing Monitoring for Intelligent Remote Management and Decision Support, Dem@Care – FP7 288199.

[5.10] B. Bishop, A. Kiryakov, D. Ognyanoff, I. Peikov, Z. Tashev, & R. Velkov, *"OWLIM: A* family of scalable semantic repositories," Semantic Web, vol. 2, no. 1, pp. 33–42, Jan. 2011

6 Conclusions

This deliverable has presented our algorithms for supporting behavioural profile learning and modelling in the scope of Dem@Care project. The contributions refer to an unsupervised framework for discovering, modelling and recognising ADL using a fixed camera, a supervised activity recognition approach for egocentric video and ontology-based patterns for capturing high-level behaviour aspects.

More specifically, we have proposed a complete unsupervised framework for discovering, modelling and recognising ADL using a fixed camera. The framework goes from low-level processing to semantic interpretation of the motion in the scene, i.e., from person tracking to inferring "preparing tea" action. Global and local human features are extracted from RGB and Depth data and they are used to learn all the meaningful regions of the scene (topologies) in an unsupervised way. Combining global and local features with topologies enables us to build primitive events in the video at three levels of resolutions. Based on these steps, we have proposed a new model for representing activities: Hierarchical Activity Model which benefits from the multiple-resolution input. This framework has been successfully tested at recognizing ADL by experimenting on patients performing semi-guided daily activities in a hospital room. We have tested our unsupervised activity recognition method using both 2D and RGB-D data. Although there are some missed activities due to failure in detecting motion, the experimental results show that the framework is a promising system that can automatically discover, learn and recognize ADL. We believe that our method can be used to study activities in home care applications and to perform fast and reliable statistics that can help doctors to diagnose diseases such as Alzheimer. As future work we plan to work on decreasing false positive and false negative rates by improving our motion and person detection algorithms, while continuing to evaluate this approach into a larger population of Dem@Care dataset.

Furthermore, a supervised activity recognition approach for egocentric video has been presented. This approach has shown very promising results with respect to the combination of two sources of information, namely active objects and location context. The active objects, either manipulated or observed by the user, provide very strong cues about the action, while the context contributes with complementary information by identifying the place in which the action is being made. Briefly, the proposed method models activities as sequences of active objects and places. An evaluation is presented for two different scenarios, where the combination of objects and context provides notable improvements in the recognition performance and outperforms state-of-the-art methods which use active and passive objects representations.

Finally, an ontology-based pattern-oriented approach for the formal modelling of user behavioural aspects has been presented. We defined a core activity pattern that extends DnS and provides the conceptual basis for structuring the individual behaviour pattern modules and six specialised patterns that allow for modelling the manner in which activities are carried out (e.g. sequence information, start/end activity patterns) as well as habitual information in terms of frequency, repetition and duration information. Next steps include the evaluation of the proposed approach within the first Dem@Care pilot context and the development of algorithms for their automated population and enrichment. Population refers to the learning of the pattern descriptions that define a given behaviour situation, namely the contextual constraints pertinent to a behaviour, e.g. making tea consists of turning on the kettle, then placing the tea bag in the cup, adding water and finally adding sugar. Enrichment on the other hand refers to the update and evolution of the behaviour situation per se, e.g. making tea now involves the addition of milk and not of sugar. Another direction would be to explore the use of behaviour patterns for activity recognition purposes.

The proposed approaches will be further evaluated in their combined form for the patient health care assessment. This evaluation would address the following questions: does the fusion of these approaches provide a richer behavioural context for the assessment of health care status? If yes, at what level should these approaches be combined? For instance, at assessment level by concurrently applying all the approaches, and then fusing their output at the end, or by only combining the most significant internal indicators of each approach on a single assessment. In this quest, the ability to capture in a formal manner patterns of higher-level behavioural aspects (e.g. the manner in which an individual makes his/her morning tea) and their dependencies with low-level behavioural traits (e.g. the use of milk or not) would enable an additional level of PwD-tailored assessment, focusing on the long-term monitoring and assessment of the behaviour.