D4.5
Activity Monitoring and Lifelogging v2

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

Dem@Care - FP7-288199

@Health
Better Healtcare 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 40, February 2015
Actual date of delivery:		Month 41, March 2015
Deliverable No.		D4.5
Deliverable Title		Activity Monitoring and Lifelogging v2
Type:		Report
Approval Status:		Approved
Version:		1.4
Number of pages:		70
WP:		WP4 Situational Analysis of Daily Activities
Task:		T4.1 Visual Perception, T4.2 Audio Sensing. T4.3. Instrumental Activities Monitoring, T4.4. Lifelogging
WP/Task responsible:		WP4/UB1
Other contributors:		INRIA, CERTH, DCU
Authors (Partner)		Duc Phu Chau (INRIA), Vincent Buso (UBX), Konstantinos Avgerinakis, Alexia Briassouli (CERTH), Feiyan Hu, Eamonn Newman (DCU),
Responsible Author	**Name**	Eamonn Newman
	Email	**eamonn.newman@dcu.ie**
Internal Reviewer(s)		Ceyhun Baruk Akgul (Vispera)
EC Project Officer		Stefanos Gouvras
Abstract (for dissemination)		Deliverable D4.5 includes the description of the final versions of Dem@Care tools for visual data analysis and their evaluation on the datasets obtained within Dem@Care for posture recognition, action recognition, activity monitoring and life-logging. D4.5 extends deliverable D4.2, which provided a study of the state of the art and a description of the first set of tools, by improving on the methods and expanding the experimental evaluation. The integration status of all visual processing components produced in this work package and their respective usage in pilots is also presented here, reflecting their impact of research outcomes in real-world Dem@Care applications.

Version Log

Version	Date	Change	Author
0.1	03/02/2015	Template for circulation	Eamonn Newman (DCU)
0.2	24/03/2015	Multi-camera tracking	Duc Phu Chau
0.3 – 0.7	27/03/2015	Initial complete template	All
0.8	28/03/2015	Technical Chapters for review	Eamonn Newman (DCU)
0.9	31/03/2015	Added Preamble, Intro, Concl.	Eamonn Newman (DCU)
1.0	15/11/2015	Added Human Activity Recognition section	Kostas Avgerinakis (CERTH)
1.1	23/11/2015	Addition of Integration and Pilot Usage Section	Thanos Stavropoulos (CERTH)
1.2	25/11/2015	Revision of entire D4.5	Alexia Briassouli (CERTH)
1.3	30/11/2015	Additional revisions to D4.5	Alexia Briassouli (CERTH)
1.4	1/12/2015	Final revisions to D4.5	Alexia Briassouli (CERTH)

Executive Summary

D4.5 presents the final version of the audio-visual analysis tools developed for Dem@Care and deployed in the integrated system and project pilots. The aim of the audio-visual analysis has been the assessment of the individual's overall status through the recognition of activities of daily living for monitoring of behavioural and lifestyle patterns, cognitive status, mood. Their integration into the final system enhances the description of the person's status and the progression of their condition due to the complementarity of the sensor data and the higher level information extraction through Semantic Interpretation (SI). The results are expected to provide new insights into dementia and its evolution over time, as well as the early detection of deterioration in the individual's status. An initial version of the tools developed within Dem@Care and the first methods used was described in D4.2, while D4.5 presents their final versions that expand and improve upon the previous ones. Finally, D4.5 presents real world experimental evaluations of tools in the integrated Dem@Care system deployed in the pilots.

This report consists of 4 main chapters, following the structure of D4.2.

Chapter 2 describes the research conducted for tracking individuals through a scene using multiple cameras, and shows how the proposed approach improves on the state of the art algorithms for single camera tracking.

Chapter 3 presents the research carried out on video analysis for Action Recognition through Object Recognition and Room Recognition on video data from a wearable camera.

Chapter 4 describes the work done for Activity Recognition and Person Detection from video and RGB-D cameras.

Chapter 5 presents Periodicity Detection on longitudinal lifelog data where signal analysis techniques are used to identify routines and periodic behaviour of an individual.

Chapter 6 presents a holistic component integration and pilot usage section, which summarizes this entire Work Package contributions of research and development, to real-world piloting and the clinical results in the context of Dem@Care.

Abbreviations and Acronyms

ADL	Activities of Daily Living
BB	Bounding Box
BoVW	Bag of Visual Words
CSV	Comma Separated Values
DTW	Dynamic Time Wrapping
EDM	Euclidean Distance Mean
FPS	Frames Per Second
GMM	Gaussian Mixture Model
GPS	Global Positioning System
HOF	Histogram of Optical Flow
HOG	Histogram of Oriented Gradients
IADL	Instrumental Activities of Daily Living
MCI	Mild Cognitive Impairment
PMVFAST	Predictive Motion Vector Field Adaptive Search Technique
PnP	Perspective - n - Point
PSD	Power Spectral Density
PwD	Person with Dementia
RANSAC	Random Sample Consensus
RGB-D	Red Green Blue - Depth
SURF	Speeded Up Robust Features
SVM	Support Vector Machine

Table of Contents

List of Figures

List of Tables

1 Introduction

The central objective of WP4 is to analyse audio-visual recordings of people with dementia so as to recognise activities and situations of interest, assess their mental and emotional state and extract behavioural and lifestyle profiles, trends, and alarms. The activities of interest and the assessment of the individuals' status are based on the clinical requirements described in D2.2 and system functional requirements presented in D7.1. The outcomes of the audio-visual situational analysis will be fused with the other sensor data for a comprehensive picture of the person's condition and its progression, and will be critical in designing and implementing the optimal approach for personalised care.

Deliverable D4.5 includes the description of the final version of Dem@Care tools for analysing visual data and their evaluation on the datasets obtained during data acquisition within Dem@Care. The visual analytics aim at posture recognition, action recognition, activity monitoring, life-logging. D4.5 extends deliverable D4.2, which presented a study of the state of the art and a description of the first set of tools, by improving upon the methods presented in it, and expanding the experimental evaluation.

D4.5 consists of 4 main chapters, following the structure of D4.2.

Chapter 2 describes the research conducted for tracking individuals through a scene using multiple cameras, and shows how the proposed approach improves upon state of the art algorithms for single camera tracking.

Chapter 3 presents the research carried out on video analysis for Action Recognition through Object Recognition and Room Recognition on video data from a wearable camera.

Chapter 4 describes the methods developed for Activity Recognition and Person Detection from video and RGB-D cameras, as well as their results on real-world recordings.

Chapter 5 presents Periodicity Detection on longitudinal lifelog data where signal analysis techniques are used to identify routines and periodic behaviour of an individual.

Chapter 6 presents a holistic component integration and pilot usage section, which summarizes all WP4 research and development contributions to real-world piloting, as well as the clinical results in the context of Dem@Care.

2 People Tracking for Overlapped Multi-cameras

2.1 Introduction

People-tracking plays an important role for activity recognition, as it allows to disambiguate between different individuals in the same scene, so as to correctly recognize the activities they are carrying out. In Dem@Care we install several cameras so as to completely cover the scene under consideration. In this deliverable, a new tracking approach using data from multiple cameras is presented. We evaluate the proposed method on the Dem@Care data and compare its results with a mono-camera tracking algorithm. The comparison shows a considerable improvement of tracking performance while using the proposed approach.

2.2 Proposed Multi-camera Tracking Approach

2.2.1 Description of the Proposed Approach

This approach comprises of three steps: mono-camera tracking, trajectory association and trajectory merging. The objective of mono-camera tracking is to compute the trajectories of each person in the scene, corresponding to each camera viewpoint. In the second step, we search for the best matching for trajectories from one viewpoint to the other. In the last step, after computing the best matching pairs, we compute the merged trajectories, taking into account the reliability of trajectories extracted from the mono-view.

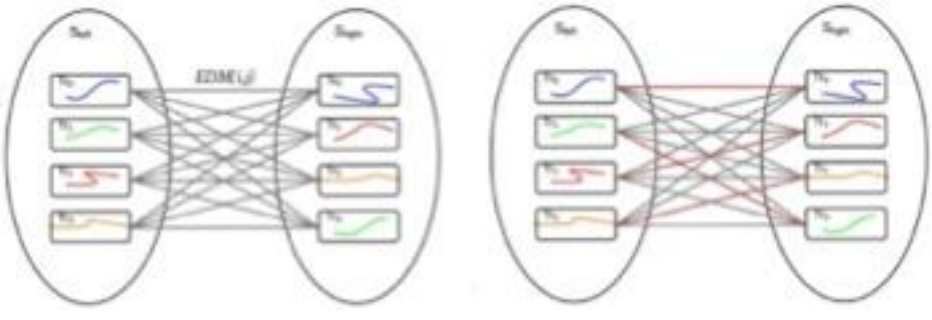
2.2.2 Mono-camera tracking

Object tracking from one camera relies on the computation of object similarity across different frames using eight different object appearance descriptors: colour histogram, colour covariance, 2D and 3D displacement, 2D shape ratio, 2D area, HOG and dominant colour. Based on these descriptor similarities, the object similarity score is defined as a weighted average of individual descriptor similarity. Trajectories are then computed as those maximizing the similarities of the objects that belong to the same trajectory.

2.2.3 Trajectory Association

The association problem is related to the need for establishing correspondences between pairwise similar trajectories that come from different cameras. The question is: which object, visible from one camera, can be associated with which objects visible from the other cameras.

For two cameras, the association or correspondence may be modelled as a bi-partite matching problem, where each set has trajectories that belong to each camera. Let C_l and C_r denote two overlapping cameras. For each camera, a set of trajectories S_{left} and S_{right} is defined. A bi-partite graph G = (V; E) is a graph in which the vertex set V can be divided into two disjoint subsets S_{left} and S_{right}, such that every edge $e \in E$ has one end point in S_{left} and the other end point in S_{right}.



(a) (b)

Figure 2-1: (a) Bi-partite graph with hypothetical associations. (b) Associations of trajectories from each camera after Hungarian algorithm is applied.

Let PO_i represent the i^{th} physical object that belongs to trajectory Tr_j^{Ck} observed by camera C_k, where $k = \{l, r\}$, n is the length of the trajectory j. Each trajectory is composed of a time sequence of physical objects:

$$Tr_j^{Ck} = \{ PO_0, PO_1, \dots, PO_i, \dots, PO_n \} \quad (1)$$

Consequently, camera C_l and C_r have a set of trajectories of size N and M called S_{right} and S_{left} :

$$S_{right} = \{ Tr_0^{Cr}, Tr_1^{Cr}, Tr_2^{Cr}, \dots, Tr_N^{Cr} \} \quad (2)$$
$$S_{left} = \{ Tr_0^{Cl}, Tr_1^{Cl}, Tr_2^{Cl}, \dots, Tr_M^{Cl} \}$$

Once the bi-partite graph is built, we need to find pair-wise trajectories similarities. To perform this task, we use spatial and temporal trajectory features. We transform the trajectory association problem across multiple cameras as follows: each trajectory Tr_j^{Ck} is a node of the bi-partite graph that belongs to set S_k for camera C_k. A hypothesized association between two trajectories is represented by an edge in the bi-partite graph, as shown in Figure 2-1(a). The goal is to find the best matching pairs in the graph.

Trajectory Similarity Calculation

There are several trajectory similarity measurements in the state of the art. We choose the Dynamic Time Warping approach (DTW) [2.11] because it is conceptually simple and effective for our trajectory similarity calculation. DTW is a dynamic-programming-based technique with $O(N^2)$ complexity, where N is the length of the trajectories to be compared. Over the last years, several authors has been studying and applying this method [2.9, 2.12, 2.13].

The NxN grid is first initialized with values of infinity (∞) that represent infinite distances. Each element (n, m) represent the Euclidean distance between two points $Tr_i^{Cl}(n)$, $Tr_j^{Cr}(m)$ $\forall n, m \in [0 \dots N]$ defined as follows:

$$d\left(Tr_i^{Cl}(n), Tr_j^{Cr}(m)\right) = \sqrt[2]{\left(x_{Tr_i^{Cl}(n)} - x_{Tr_j^{Cr}(m)}\right)^2 + \left(y_{Tr_i^{Cl}(n)} - y_{Tr_j^{Cr}(m)}\right)^2} \quad (3)$$

where (x, y) is the 2D location of trajectories after projecting on a reference view.

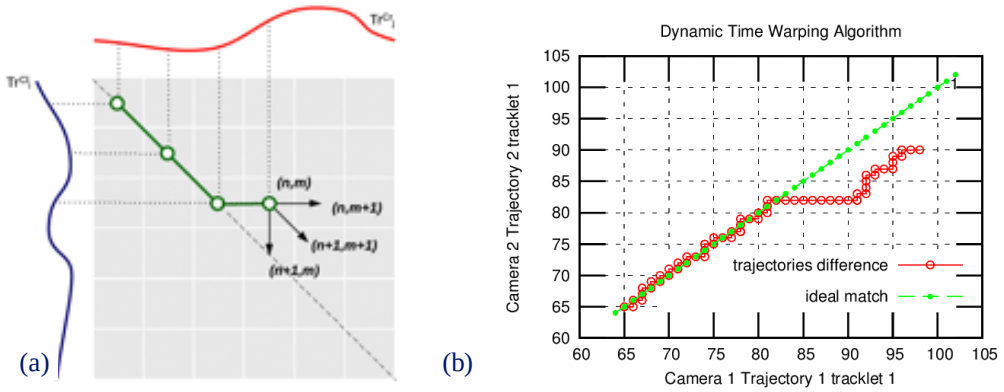


Figure 2-2: (a) The optimal warping path. (b) DTW results for tracklet 1 of two trajectories comparison. In X and Y the frames are shown. The optimal path is represented in green, and the DTW result is shown in red.

Many paths connecting the beginning and the ending point of the grid can be constructed. The goal is to find the optimal path that minimizes the global accumulative distance between both trajectories.

From the DTW results, we build a cost matrix with a normalized Euclidean Distance Mean based metric for each trajectory pair. In order to normalize the distance values computed by DTW, we divide a distance value by the maximum possible distance between two trajectories which is the diagonal of the image:

$$EDM(i,j) = \frac{D\left(Tr_i^{Cl}, Tr_j^{Cr}\right)}{\sqrt[2]{\left(\left(Image_{Height}\right)^2 + \left(Image_{Width}\right)^2\right)}} \qquad (3)$$

Now the bi-partite graph is complete and the weight of each edge $e \epsilon E$ in $G = (V; E)$ is given by EDM(i, j) (Figure 2-1(a)). The task at hand now consists in finding the optimal matching of G, aimingto find the optimal assignment that maximizes the total cost of a matrix. To find the optimal matching in G we apply the Hungarian Algorithm defined by Kuhn [2.15], given the cost matrix built with the EDM values. The Hungarian method is a combinatorial optimization algorithm that solves the assignment problem in polynomial time $O(n^3)$, where n is number of nodes or vertexes V of the bi-partite graph G. After apply the Hungarian Algorithm to the matrix we got the maximum matching as is shown in Figure 2-1(b).

2.2.4 Trajectory Merging

Once association is done, the next step is to compute the final trajectory by merging corresponding trajectories from each view. To merge two trajectories coming from two different cameras, e.g. $Tr_i \in S_{right}\ with\ 0 < i < N$ and $Tr_j \in S_{left}\ with\ 0 < j < M$ into a global one Tr_{Gij} , we apply an adaptive weighting method as follows:

$$Tr_{G_{i,j}}(t) = \begin{cases} w_1 Tr_i^{C1}(t) + w_2 Tr_j^{C2}(t) & iff\ Tr_i^{C1}(t), Tr_j^{C2}(t)\ \exists\ t \\ Tr_i^{C1}(t) & iff\ Tr_i^{C1}(t) \exists\ t \wedge Tr_j^{C2}(t)\ \nexists\ t \\ Tr_j^{C2}(t) & iff\ Tr_j^{C2}(t) \exists\ t \wedge Tr_i^{C1}(t)\ \nexists\ t \end{cases} \qquad (4)$$

As we defined in Eq. (3), each trajectory is composed by a set of detections. Chau et *al.* [2.1] defined a method to quantify the reliability of the trajectory of each interest point by considering the coherence of the Frame-to-Frame (F2F) distance, the direction, and the HOG similarity of the points belonging to a same trajectory. Thus, as each physical object has reliability attribute R with values [0, 1]. The weight of each trajectory is defined in term of its R-value as follows:

$$w_1 = \frac{R_{PO_n}}{R_{PO_n} + R_{PO_m}} \qquad w_2 = \frac{R_{PO_m}}{R_{PO_n} + R_{PO_m}} \qquad (5)$$

It is important to note that $w_1 + w_2 = 1$. The merged trajectory is located in between the two cameras from mono-view, and is closer to the trajectory with higher reliability values.

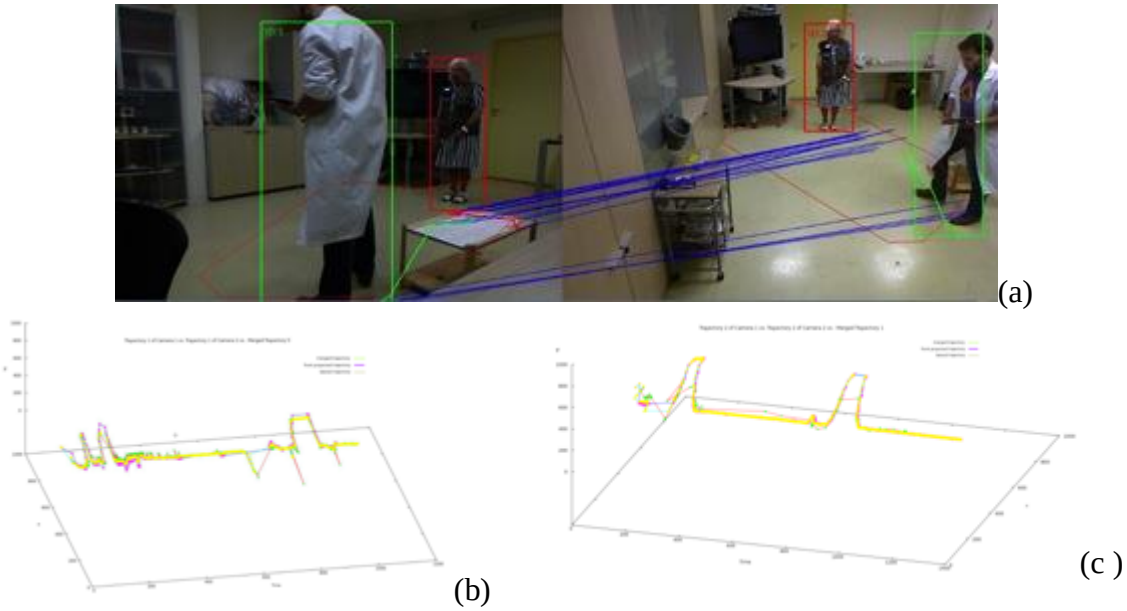
2.3 Discussion and results

The objective of this evaluation is to prove the effectiveness of the proposed approach, and to compare it with a mono-camera tracking. We select five videos from the Dem@Care dataset recorded in the Centre Hospitalier Universitaire Nice (CHUN) hospital, which involved participants with dementia over 65 years old. Experimental recordings used two widely separated RGB-D cameras (Kinect®, Microsoft©) with 640x480 pixel resolution, recording between 6 and 9 frames per second. Each pair of videos has two different views of the scene, lateral, and frontal, with two people per view, the person with dementia and the doctor. They sometime cross each other or are hidden behind furniture. They exit the scene and re-enter it several times.

Figure 2-3(a) shows two camera views of the scene. The blue lines represent the trajectory projection from the left camera to the right camera, which has been selected as reference. After the whole video is processed, we obtain the trajectories association and fusion for the doctor and patient trajectories. In *Figure 2-3*(b) the trajectory (x, y) in terms of the time (in frames) is presented. The yellow is the final patient trajectory, which is in between the right camera trajectory, and the projection of the left camera trajectory. *Figure 2-3*(c) presents the doctor's trajectory with the same colour annotation.

In order to quantify our results, we use the tracking time-based metrics from [2.18]. The tracking results are compared against entire trajectories of ground truth data. This metric gives us a global overview of the performance of the tracking algorithm. In this section we present the overall evaluation of our multi-camera tracking approach and its comparison with the mono-camera tracking algorithm [2.1].

Table 2.1 presents the tracking results of the proposed approach and mono-camera tracking approach resulting from the right viewpoint of a video. Our multi-camera tracking approach provides much better performance compared to the mono-camera tracking approach. Tracking time increases 20.79% for the first trajectory (doctor), and 6.41% for the second one (patient).



(a) (b) (c)

Figure 2-3: (a) The doctor and patient projections from the left view to the right. (b) Patient trajectories from mono-view and after merging. (c) Doctor trajectories from mono-view and after merging.

Table 2.1: Mono and multi-camera tracking results for the right camera view for a video

Approaches	Camera view	Object 1 (Doctor)	Object 2 (Patient)
		Tracking time	Tracking time
Mono-camera tracker	Right	49,67%	86,31%
Our multi-camera tracker	Left and Right	**70,45%**	**92,72%**

Table 2.1 presents the tracking results of the proposed approach and mono-camera tracking approach resulting from the left and right views of 5 videos (10 people in total). We achieved considerably better performance than the mono-camera tracking algorithm of [2.1]. The multi-camera tracking approach outperforms the mono-camera results for both camera views. For the doctor's trajectory, the most significant improvement is against the right camera viewpoint's result, which is surpassed by 19.67%. In the case of the patient's trajectory the best results (an improvement of 25.5%) are achieved compared to the results from the left camera viewpoint. For the person with dementia we achieved a high tracking time, of 91.3%, but only 66.92%for the doctor trajectory, which can be attributed to misdetection of the doctor from both viewpoints.

Table 2.2: Tracking results of the proposed approach and mono-camera tracking approach resulting from left and right view for 5 videos

Approaches	Camera view	Object 1 (Doctor)	Object 2 (Patient)
		Tracking time	Tracking time
Mono-camera tracker	Right	47,24%	85,21%
Mono-camera tracker	Left	52,76%	65,79%
Our multi-camera tracker	Left and Right	66,92%	91,30%

2.4 Conclusion

We have presented a novel multi object tracking process for multiple cameras. For each camera, tracking by detection is performed. Trajectory similarity is computed using a Dynamic Time Warping approach. Afterwards, the association of trajectories takes place as a maximum bi-partite graph matching, addressed by the Hungarian algorithm. Finally, the merging processes between associated trajectories has taken place with an adaptive weighting method.

We evaluate the multi-camera approach in a real-world scenario and compare its results to the mono-camera approach. Our method considerably outperforms the mono-camera tracking algorithm [2.1], with good occlusion management and providing more complete trajectories by recovering additional information, which is not available in a single view. In future work, we will modify this algorithm to achieve online processing.

2.5 References

[2.1] D. P. Chau, F. Bremond, and M. Thonnat, “A multi-feature tracking algorithm enabling adaptation to context variations,” The International Conference on Imaging for Crime Detection and Prevention (ICDP), 2011. [Online]. Available: http://arxiv.org/pdf/1112.1200v1.pdf. [Accessed: 25-Apr-2014].

[2.2] N. Anjum and A. Cavallaro, “Trajectory Association and Fusion across Partially Overlapping Cameras,” 2009 Sixth IEEE Int. Conf. Adv. Video Signal Based Surveill., 2009.

[2.3] Y. A. Sheikh and M. Shah, “Trajectory association across multiple airborne cameras.,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 30, pp. 361–367, 2008.

[2.4] T.-H. Chang and S. Gong, “Tracking multiple people with a multi-camera system,” Proc. 2001 IEEE Work. Multi-Object Track., 2001.

[2.5] S. M. Khan and M. Shah, “A Multiview Approach to Tracking People in Crowded Scenes Using a Planar Homography Constraint,” Comput. Vision–ECCV 2006, vol. 3954, pp. 133–146, 2006.

[2.6] R. Eshel and Y. Moses, “Homography based multiple camera detection and tracking of people in a dense crowd,” 2008 IEEE Conf. Comput. Vis. Pattern Recognit., 2008.

[2.7] Richard Hartley and Andrew Zisserman, Multiple View Geometry in Computer Vision. Cambridge university press, 2003.

[2.8] A. Yilma and M. Shah, "Recognizing human actions in videos acquired by uncalibrated moving cameras," Tenth IEEE Int. Conf. Comput. Vis. Vol. 1, vol. 1, 2005.

[2.9] Peng Chen, Junzhong Gu, Dehui Zhu, Fei Shao, "A Dynamic Time Warping based Algorithm for Trajectory Matching in LBS," Int. J. Database Theory Appl., vol. Vol. 6, no. Issue 3, pp. p39–48. 10p., 2013.

[2.10] L. Bergroth, H. Hakonen, and T. Raita, "A survey of longest common subsequence algorithms," Proc. Seventh Int. Symp. String Process. Inf. Retrieval. SPIRE 2000, 2000.

[2.11] A. Kassidas, J. F. Macgregor, and P. A. Taylor, "Synchronization of batch trajectories using dynamic time warping," AIChE J., vol. 44, pp. 864–875, 1998.

[2.12] H. J. Ramaker, E. N. M. Van Sprang, J. A. Westerhuis, and A. K. Smilde, "Dynamic time warping of spectroscopic BATCH data," Anal. Chim. Acta, vol. 498, pp. 133–153, 2003.

[2.13] Y. Z. Y. Zhang and T. F. Edgar, "A robust Dynamic Time Warping algorithm for batch trajectory synchronization," 2008 Am. Control Conf., 2008.

[2.14] T. Kashima, "Average trajectory calculation for batch processes using Dynamic Time Warping," SICE Annu. Conf. 2010, Proc., 2010.

[2.15] H. W. Kuhn, "The Hungarian Method for the Assignment Problem," 50 Years Integer Program. 1958-2008, pp. 29–47, 2010.

[2.16] K. Bernardin and R. Stiefelhagen, "Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics," EURASIP J. Image Video Process., vol. 2008, pp. 1–10, 2008.

[2.17] Y. Li, C. Huang, and R. Nevatia, "Learning to associate: HybridBoosted multi-target tracker for crowded scene," 2009 IEEE Conf. Comput. Vis. Pattern Recognit., 2009.

[2.18] A. T. Nghiem, F. Bremond, M. Thonnat, and V. Valentin, "ETISEO, performance evaluation for video surveillance systems," in Advanced Video and Signal Based Surveillance, 2007. AVSS 2007. IEEE Conference on, 2008, pp. 476–481.

3 Action Recognition

3.1 3D Localization From Wearable Camera

In this work, we are interested in estimating the position of a patient moving in an apartment from a wearable camera (see Fig. 3-1). To achieve this, we first develop a complete 3D reconstruction framework to robustly and accurately reconstruct a whole apartment from a training video. Then, we propose a localization approach that relies on the 3D model of the environment to estimate the position of the camera from a new video.

Figure 3-1: 3D Localization From Wearable Camera Problem

3.1.1 Reconstruction of an apartment from a wearable camera

In D4.3 we presented a 3D reconstruction framework that is able to estimate both the camera pose, as well as a sparse 3D point cloud from a few hundred images of a single room. However, when the number of images increases, the computational complexity of the approach quickly becomes prohibitive. Thus the previously proposed framework can only reconstruct a room and not a whole apartment. Here, we are interested in building a globally consistent 3D model of an entire apartment.

Our new large-scale 3D reconstruction framework builds upon the framework introduced in the previous deliverable. It also deals with videos and thus takes advantage of the temporal continuity of the video frames, while the previous framework assumed that the frames where unordered. The workflow of the proposed approach is illustrated in Figure 3-2.

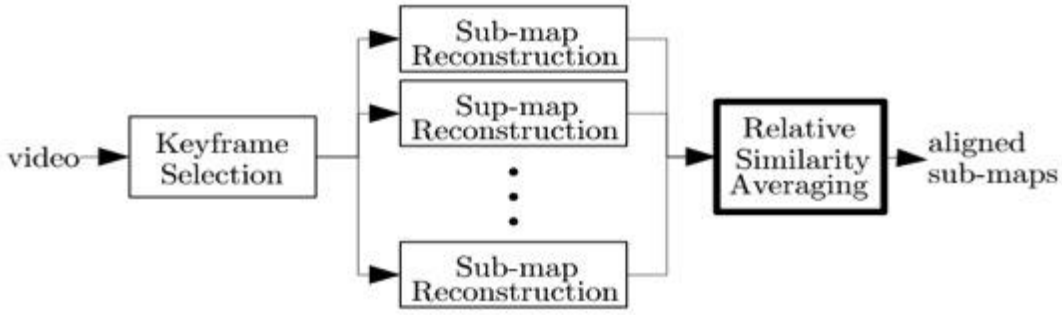


Figure 3-2: Workflow of the proposed large-scale 3D reconstruction framework

3.1.2 Keyframe Selection

First of all, our approach selects keyframes among all the video frames by running a Lucas-Kanade tracker on the frames. It works as follows:

[1] The first video frame is a keyframe.

[2] Iterate until the end of the video

[1] Set next video frame as current frame

[2] Run the Lucas-Kanade tracker

[3] If the distance between the 2D points tracked in the previous keyframe and the 2D points matched to them in the current frame is higher than a threshold (typically 10% of the width of a frame) then set the current frame as a keyframe

[4] Go to 1

This keyframe selection allows us to take advantage of the temporal continuity of the video frames by *tracking* features between keyframes instead of simply trying to *match* features between keyframes.

3.1.3 Sub-map Reconstruction

After having selected keyframes, we define overlapping subsets of consecutive keyframes. For each subset of keyframes, we estimate a sub-map, i.e a 3D point cloud as well as the camera poses, with a framework similar to the one described in the previous deliverable. The workflow of the sub-map reconstruction framework is illustrated in Figure 3-3. The main modification with respect to the previous deliverable relies in the modification of the "global camera orientation estimation" where we now employ an iterated extended Kalman filter on Lie groups (accepted in ICIP 2014). Further details on this sub-map reconstruction framework have been submitted to CVPR 2015 and will be available soon.

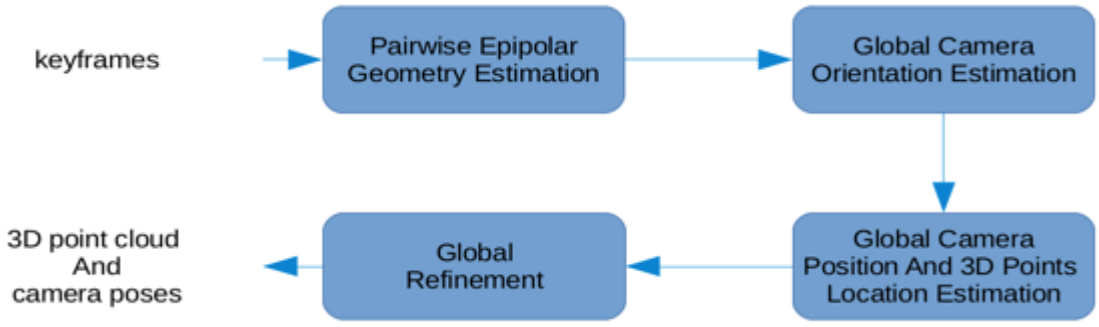


Figure 3-3: Sub-map reconstruction framework

3.1.4 Relative Similarity Averaging

Once all the sub-maps have been estimated, we need to align them to obtain a globally consistent 3D model. Aligning these sub-maps consists in computing and averaging relative 3D similarities (scale, rotation and translation) between them.

To compute the relative similarities, we propose the following approach:

For each point cloud:

- Select the 3D points that have a "small " covariance.
- Compute a signature. We use a bag of visual words approach and consider the histogram as a signature.
- Find the 100 "closest" point clouds. Here "closest" means w.r.t the L2 distance between signatures.
- Match the 3D point descriptors of each of these point clouds to the 3D points descriptors of the current point cloud.
- Compute the relative 3D similarities between the best 30 point clouds and the current point cloud by minimizing the distances between the matched 3D points. If two sub-maps are overlapping, i.e. if they share cameras, then we also include the distance between these cameras poses in the criterion. In this step, a RANSAC algorithm is applied since matches between 3D points usually produces outliers.

Once that all relative 3D similarities have been computed, we need to average them. To do so, we apply the iterated extended Kalman filter on Lie groups (that was accepted in ICIP 2014).

Unfortunately, this algorithm is not robust to outlier measurements while the relative 3D similarities might contain outliers. Indeed, two point clouds representing two places from different rooms that have a similar geometry might produce a relative similarity measurement. In order to deal with these outliers, we apply a robust approach, still based on the iterated extended Kalman filter on Lie groups that was accepted in ACCV 2014.

As we will see in the next section, this approach is able to efficiently reject outliers, while aligning the sub-maps to obtain a globally consistent 3D model.

3.1.5 Qualitative Results

We qualitatively evaluate the performance of our system on a video sequence of 10000 frames recorded in an apartment. Our framework currently runs in Matlab and took 2.5 hours

to process the training video. Even if this is already a reasonable computation time, it could be significantly reduced using C/C++. After having applied our automatic framework to the video sequence, we obtained a set of aligned sub-maps, with each sub-map containing a 3D point cloud and part of the camera trajectory. In order to qualitatively evaluate the result, we manually place the estimated camera trajectory from the training video sequence on top of the ground plan of the flat (see Figure 3-4). As it can be seen, the superimposed camera trajectory is coherent with the ground plan, i.e it trajectory goes into (almost) every room without crossing the walls and passes through the doors when going from one room to another.

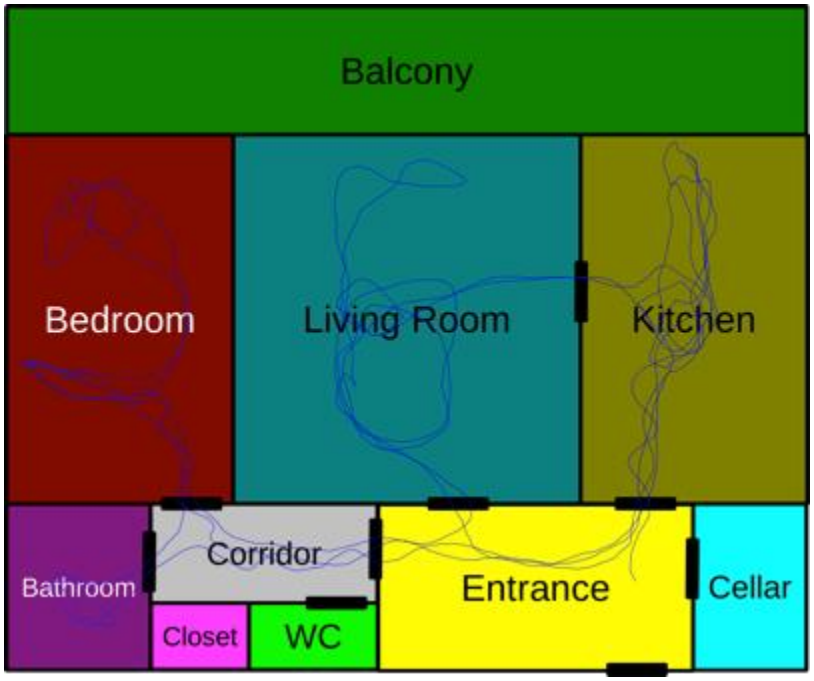


Figure 3-4: Superimposed reconstructed camera trajectory (blue line) with the ground plan of the flat.

3.1.6 Metric Localization from a wearable camera

In the previous section, we presented a new algorithm to reconstruct a 3D model of an entire apartment from a training video. We now propose a localization framework that relies on this 3D model to estimate the position of the camera from a new video. This new localization algorithm employs two different place detectors. The first place detector is based on the appearance of the scene, which provides robustness to motion blur and moving objects. The second detector is based on the 3D geometry of the model, which makes it highly accurate but not always available. The result of both detectors are fused using a novel Rao-Blackwellized particle filter on the Lie group SE(3) that relies on a white noise acceleration model to produce the final camera trajectory. The proposed framework is presented in Figure 3-5.

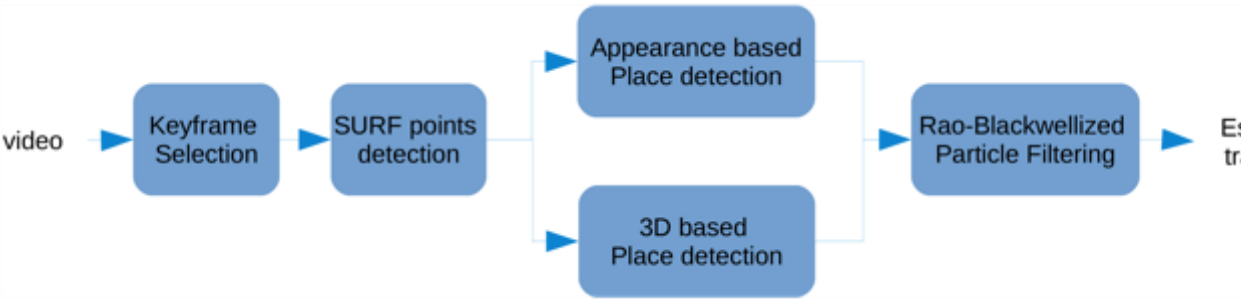


Figure 3-5: Localization framework

Keyframe Selection

In order to reduce the number of frames to localize, we perform keyframe selection over the video frames to only keep the frames where the person is actually moving. This keyframe selection step is the same as the one depicted in the previous section.

SURF points detection

After having selected keyframes, SURF points are extracted. These points will be used both by the appearance based place detector and the 3D based place detector.

Appearance based place detection

For each keyframe of the 3D model previously reconstructed, we have saved its SURF points, a 32x32 miniature as well as its position and orientation w.r.t the 3D point cloud. The aim of this module is to exploit the information, that we call the "appearance" of the 3D model, to localize each keyframe of the new video. Here, we do not use the reconstructed 3D point cloud.

We now detail how a keyframe K of the new video is localized.

First of all, a 32x32 miniature is created and compared to the miniatures of the 3D model using the L1-norm. Then, the 100 closest keyframes of the 3D model are re-ranked by matching their SURF points to those of K using kd-trees and bi-directional matching.

Finally, from the 10 closest keyframes of the 3D model, a mixture of Gaussian distributions is created where the mean of each component is set as the pose of the corresponding keyframe, the covariance is defined by hand (each component has the same covariance) and the weight is proportional to the number of matches. This mixture of Gaussians will be used by the Rao-Blackwellized particle filter.

3D based place detection

The aim of this module is to exploit the reconstructed 3D point cloud to localize each keyframe of the new video.

To do so, for a keyframe K, we first match its SURF points to the SURF points of the 3D model. We then apply a PnP algorithm combined with a RANSAC to robustly estimate the pose of K. Based on a Gauss-Newton algorithm, the estimated pose is finally refined by minimizing the reprojection error of the 3D points in the keyframe. Around the estimated

pose, the covariance matrix of the estimated errors is approximated by performing a Laplace-like approximation.

To increase the performances of this 3D based detector, the described method is actually applied to the point cloud of each sub-map.

The output of this detector is thus once again a mixture of Gaussian distributions that will be used by the Rao-Blackwellized particle filter.

Rao-Blackwellized Particle Filter on SE(3)

The aim of this module is to fuse the information coming from the appearance-based place detector, which is robust to motion blur and moving objects, and the 3D based place detector which is highly accurate but not always available, to reliably estimate the camera trajectory.

At each time instant, a mixture of Gaussian distributions (from the two place detectors) is provided to the filter to select the "true" component. To do so, the filter employs spatiotemporal *a priori* information, which states that camera poses should be close to each other for two consecutive time instants. We use a white noise acceleration motion model to represent this *a priori* information.

Consequently, at each time instant, the filter estimates the component of the mixture to select as well as the pose of the camera (and its speed).

The discrete part of the state (the component selection) is sampled while the continuous part (camera pose and speed) is solved analytically using an iterated extended Kalman filter on Lie groups (ICIP 2014).

Qualitative and quantitative results

In order to evaluate the proposed localization framework, we recorded several videos in the apartment that we previously reconstructed and manually built ground truth trajectories. Then we applied the proposed localization framework to estimate camera trajectories. In Figures 3-6 and 3-7, estimated camera trajectories and ground truth trajectories are represented for two sequences. In the following table, the average position error is represented for these videos.

	Sequence 1	Sequence 2
Average Position Error (m)	0.54	0.64

One can see, that for those two videos, the camera trajectory is accurately estimated. The current Matlab implementation achieves 1.3 FPS.

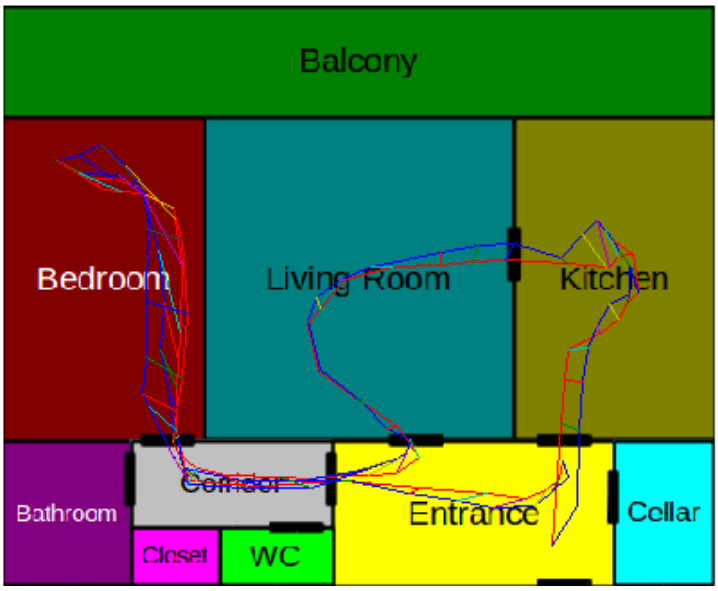


Figure 3-6: Sequence 1 - Superimposed estimated camera trajectory (blue line) with the ground truth (red line).

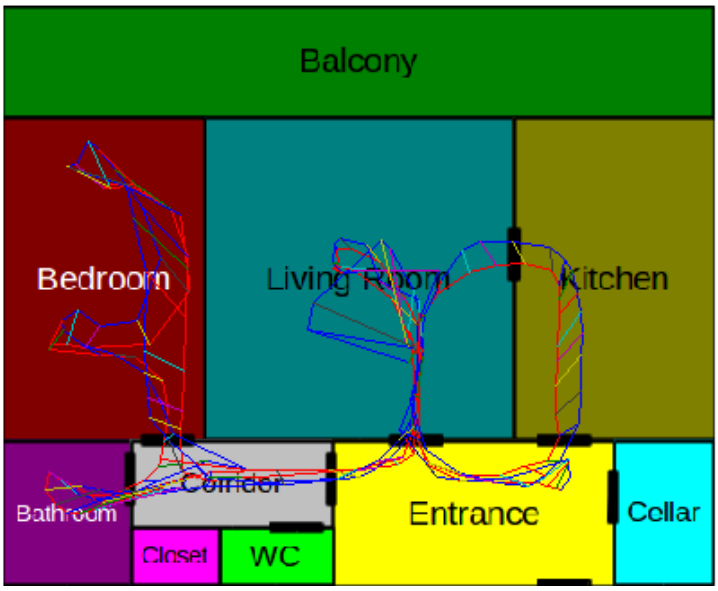


Figure 3-7: Sequence 2 - Superimposed estimated camera trajectory (blue line) with the ground truth (red line)

3.1.7 Conclusion and Future Work

In this work, we first presented a complete 3D reconstruction framework able to robustly and accurately reconstruct a whole apartment from a training video. Then, we proposed a localization approach that relies on the 3D model of the environment to estimate the position of the camera from a new video. We demonstrated, both qualitatively and quantitatively, that the proposed localization framework was able to accurately estimate the camera trajectory from a new video in an apartment previously reconstructed with the proposed 3D

reconstruction framework. As future work, we are interested in cases where the place detectors fail, for example when someone puts their hand in front of the camera.

3.2 Object recognition

3.2.1 Objectives

For the task of the assessment and life-logging of Alzheimer patients in their Instrumental Activities of Daily Living (IADLs), egocentric video analysis has gained strong interest as it allows clinicians to monitor patients' activities and thereby study their condition and its evolution and/or progression over time. Recent studies demonstrated how crucial the recognition of manipulated objects is for activity recognition under this scenario [3.1, 3.2].

As described in earlier deliverables D4.1, D4.3, D4.4, visual saliency is an efficient way to drive the scene analysis towards areas 'of interest' and has become very popular among the computer vision community. Manipulated object recognition tasks can greatly benefit from visual attention maps both to reduce the computational burden and filter out the most relevant information.

Generally speaking, two types of attention are commonly distinguished in the literature: bottom-up or stimulus-driven and top-down attention or goal-driven [3.3, 3.4]. The authors of [3.3] define the top-down attention as the voluntary allocation of attention to certain features, objects, or regions in space. They also state that attention is not only voluntary directed as low-level salient stimuli can also attract attention, even though the subject had no intention to attend these stimuli. A recent study [3.5] about how saliency maps are created in the human brain, shows that an object captures our attention depending both on its bottom-up saliency and top-down control.

Modelling of human visual attention has been an intensively explored research subject since the last quarter of the 20th century and nowadays the majority of saliency computation methods are designed from a bottom-up perspective [3.6]. Bottom-up models are stimulus-driven, mainly based on low-level properties of the scene such as color, gradients orientation, motion or even depth. Consequently, bottom-up attention is fast, involuntary and, most likely feed-forward [3.6].

However, although the literature concerning models of top-down attention is clearly less extensive, the introduction of top-down factors (e.g., face, speech and music, camera motion) into the modelling of visual attention has provided impressive results in previous works [3.7, 3.8]. In addition, some attempts in the literature have been made to model both kinds of attention for scene understanding in a rather "generic" way. In [3.9] the authors claim that the top-down factor can be well explained by the focus in image, as the producer of visual content always focuses his camera on the object of interest. Nevertheless, it is difficult to admit this hypothesis for expressing the top-down attention of the observer of the content: it is always task-driven [3.6].

More recent works using machine learning approaches to learn top-down behaviours based on eye-fixation or annotated salient regions, have proven also to be very useful for static images [3.10, 3.11, 3.12] as well as for videos [3.13, 3.14]. Furthermore, with advent of Deep Learning Networks (DNN), some novel approaches have been designed in the field object recognition, which build class-agnostic object detectors to generate candidate salient bounding-boxes which are then labelled by later class-specific object classifiers [3.15, 3.16]. However, it seems impossible for us to propose a universal method for prediction of the top-

down visual attention component, as it is voluntary directed attention and therefore it is specific for the task of each visual search. Nevertheless, the prior knowledge about the task the observer is supposed to perform, allows extracting semantic clues from the video content that ease such a prediction.

The current state-of the art in computer vision allows for the detection of some categories of objects with high confidence. A variety of face or skin detectors have been proposed in the last two decades [3.17]. Hence, when modelling top-down attention in a specific visual search task, we can use "easily recognizable" semantic elements that are relevant to the specific task of the observer and may help to identify the real areas/objects of interest.

In this work we propose to use domain specific knowledge to predict top-down visual attention in the task of recognizing manipulated objects in egocentric video content. In particular, our "recognizable elements" (those relevant to the task) are the arms and hands of the user wearing the camera and performing the action. Their quantized poses with regard to different elementary components of a complex action such as object manipulation will help in the definition of the area where the attention of the observer searching for manipulated objects will be directed.

We evaluate our model from two points of view: i) prediction strength of gaze fixations of subjects observing the content with the goal of recognition of a manipulated object, and ii) performance in the target object recognition by a machine learning approach.

3.2.2 Goal-oriented top-down visual attention model

Our new top-down model of visual attention prediction in the task of manipulated object recognition relies on the detection and segmentation of some objects, considered as references, that help locate the actual areas of interest in a scene, namely the objects being manipulated. In our proposal, arms/hands are automatically computed for each frame using the approach introduced by Fathi et al. [3.18].

We propose to build our model as a combination of two distinct sets of features: global and local. The former describes the geometric configuration of the segmented arms, which are clustered into a pre-defined set of states/configurations. This global information is used to select one of the components in a mixture model. The second set, concerning the local features, is then modelled using the specific distributions corresponding to the selected global component.

Defining Global features

The features we propose are based on the geometry of arms in the camera field of view, which is correlated with the manipulated objects' size and position. An elliptic region in the image plane approximates each arm, from the elbow to the hand extremity. Hence an ellipse is first fitted to each segmented arm area and, then, several global features are defined, namely:

- *Relative location of hands*: Two features are extracted that encode the relative location of one hand with respect to the other (see Figure *3-8* (a)). To that end, and when taking the left hand centre as the origin of coordinates, the vector that joins the origin and the right hand is represented by means of its magnitude ρ_{Rel} and phase φ_{Rel}. The

magnitude and phase are strong indicators of the objects' width and holding pose, respectively.

- *Left arm orientation* and *Right arm orientation*: As illustrated in Figure *3-8* (b) the orientation of each arm φ_L and φ_R is defined by the angle between principle axis of the ellipse and the Y-axis in an image plane. The arms are mostly oriented depending on the objects being manipulated, e.g. holding a cup or pouring something (e.g., milk, juice) usually present distinguishable arm orientations.
- *Left arm depth* and *Right arm depth with regard to the camera*: an object size is likely to be correlated with the "depth" of the arms, i.e. a measure of its closeness to the camera. In this work, body-worn cameras do not provide real depth information. A trivial approximation of the "depth" of an arm is the minor axis length d_L and d_R) of the fitted ellipse (see Figure *3-8* (c)).

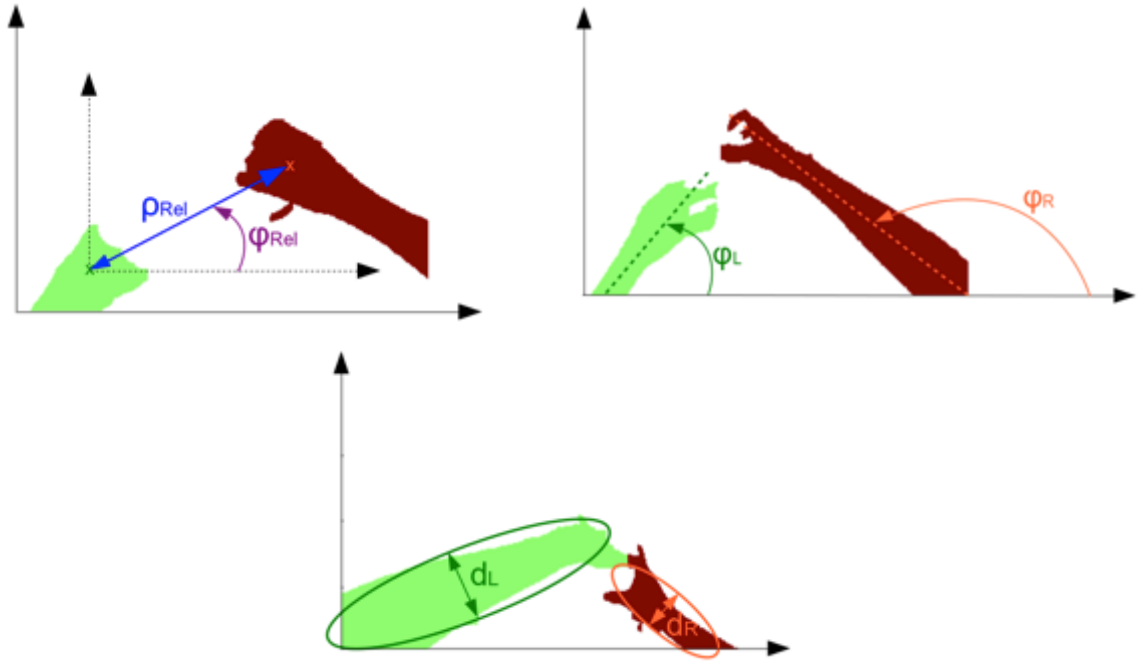


Figure 3-8 Illustrations of the 6 global features. 1(a): Relative location of hands, 1(b): Left arm orientation, 1(c): Left arm depth and Right arm depth with regard to the camera.

A vector $\boldsymbol{g} = (\rho_{Rel}, \varphi_{Rel}, \varphi_L, \varphi_R, d_L, d_R)$ containing these six geometrical features is computed for each image in the training set, and then clustered into K global appearance models using k-means. It is worth noting that Z-score normalization has been performed over the data, in order to prevent outweighing features with large range over attributes with small ones [3.19]. Figure 3-9 illustrates results in case of 8 clusters in our training dataset. The difference between the global appearance states (a) - (h) is easily noticeable.

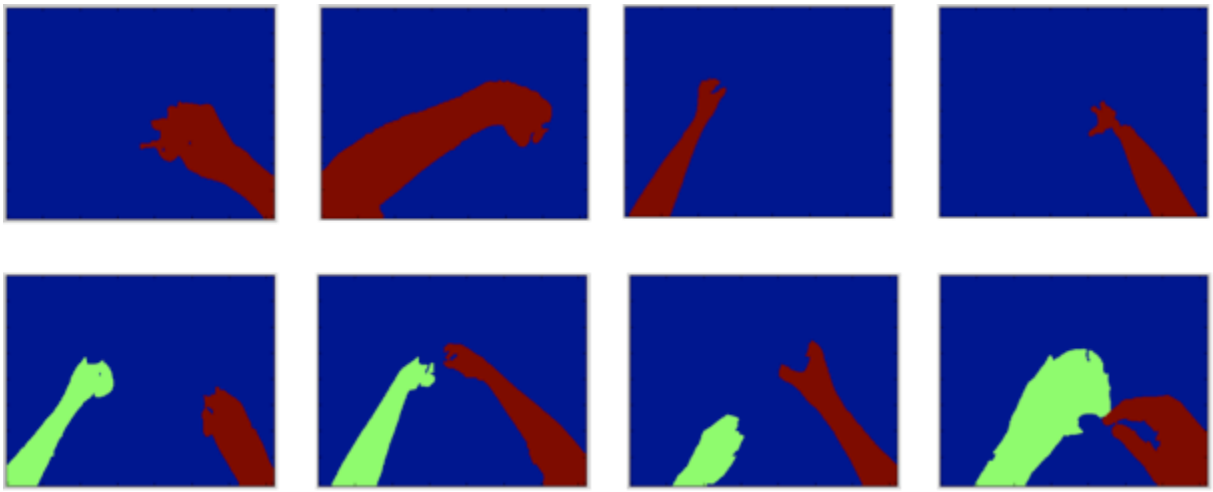


Figure 3-9 Representation of the arm segmentations closest to the centre of 8 global appearance model clusters. Each cluster is represented by the sample that is closest to the cluster centre.

Defining Local Features

Global appearance models define the most common states in which the arms can be found. Depending on these models, the zones of interest are different and the saliency computation needs to be adapted to. The "local" features we introduce serve for refining the underlying saliency distribution in the frame for a given global state. These features are the coordinates of a hand centre $\boldsymbol{c}$ (or hand centres in case the global state contains two hands). Their computation is also based on geometrical considerations.

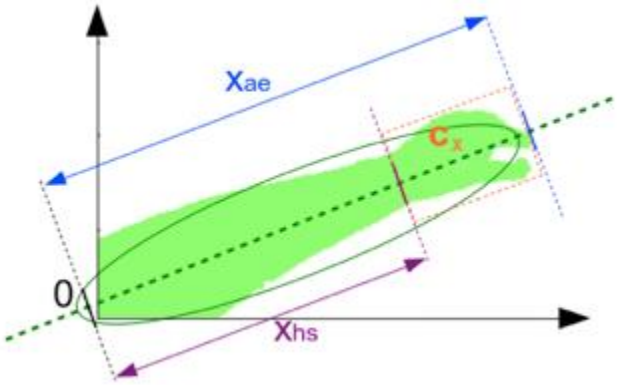


Figure 3-10 Illustration of the hand centre c computed as the barycentre of the orange box and the key points around: x_{hs} the starting position of the hand on the major ellipse axis, and x_{ae} the end position of the whole arm.

Intuitively, when only the hand appears in the image, the hand centre $\boldsymbol{c}$ should be situated around the center of mass of the entire segmented image. Similarly, if the whole arm appears as in Figure 3-10, the hand centre should be located closer to the extremity of the arm. Looking at Figure 3-10, let us define two segments: x_{hs} is the segment that joins the beginning of the arm (origin of coordinates) with the beginning of the hand, and x_{ae} is the full arm-length. We have observed that the ratio $d = \frac{x_{hs}}{x_{ae}}$ is closely related with the ratio r between the minor and

major axis of the fitted ellipse. In particular, to establish this relationship, we have randomly select some training frames for which we annotated the hands starting points x_{hs} over the major axis of the ellipse (represented as blue dots in Figure 3-11), and then optimized an exponential model as:

$$d(r) = a\, e^{br}$$

where a and b are the coefficients computed by exponential fitting, r is the ratio between minor/major axis of the ellipse fitting the segmented arm, and $d(r)$ gives the ratio between the starting points x_{hs} of hands on the major ellipse axis and the arm length x_{ae} as a function of r. The results of this optimization are shown (red line in Figure 3-11).

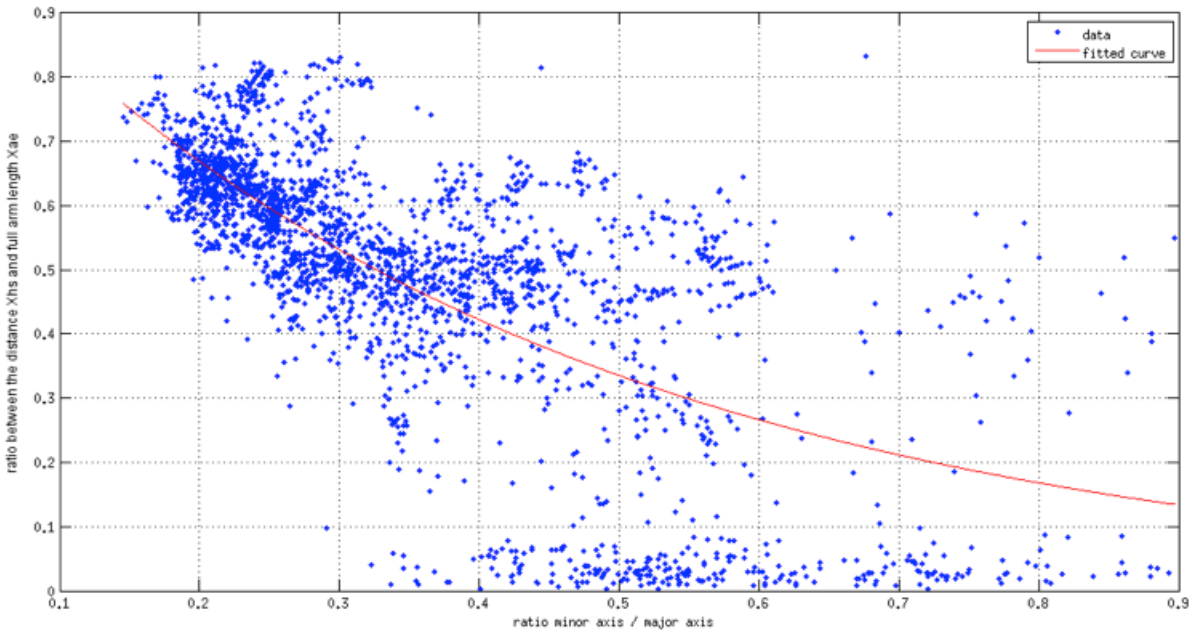


Figure 3-11 Graph representing the ratio between hands beginning and arm length depending on the minor/major axis lengths of the ellipse fitting the segmented arms. Blue dots correspond to the values manually annotated, red line to the fitting exponential model

Finally, the two-dimensional center $\boldsymbol{c}$ coordinates are then defined as the center of mass of the segmented area that lies between the starting point of the hand and the end of the arm (the center of the orange box in Figure 3-10. The computed "hand centre" $\boldsymbol{c}$ coordinates will generate the hand-related saliency map.

A Probabilistic Model for Top-down Visual Attention Prediction

As a human observer would be attracted to the objects manipulated by hands, we consider the joint locations of arms/hands and objects as predictors of top-down visual attention. Hence, we have developed a probabilistic model for top-down visual attention that incorporates both global and local features distributions. The graphical model of our approach for Top-Down

visual attention is shown in Figure 3-12. Given a corpus of D training images, the objective is to learn the process that chooses a set of N salient spatial locations $\boldsymbol{x}$ within each frame.

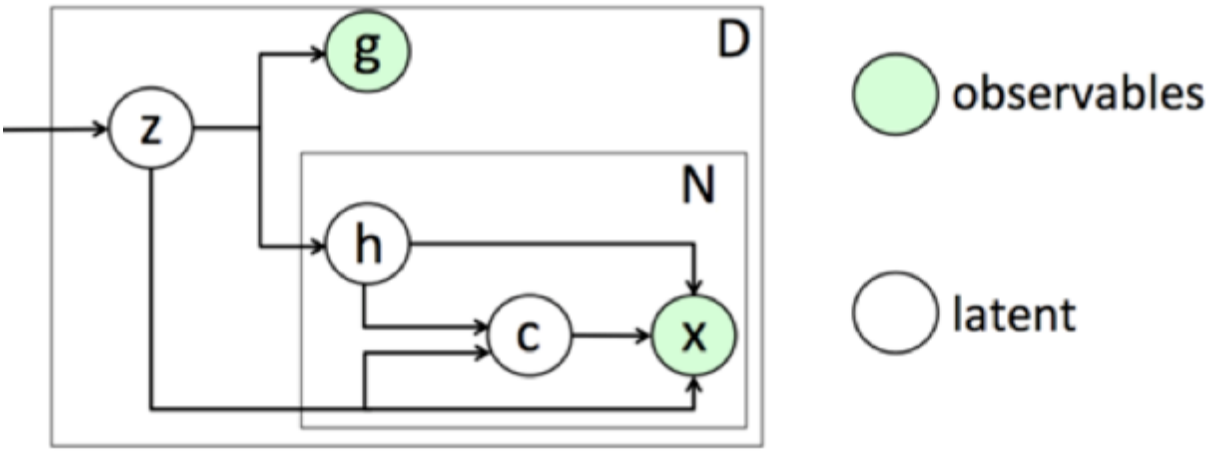


Figure 3-12 Graphical model of our approach Top-down visual attention modelling with manipulated objects. Nodes represent random variables (observed-shaded, latent-unshaded), edges show dependencies among variables, and boxes refer to different instances of the same variable.

Let us first introduce a simplified model considering just the set of K global arm configurations $\boldsymbol{z} = \{z_1, \dots, z_K\}$, and their relationship with the global features $\boldsymbol{g}$. Given $\boldsymbol{z}$, the probability density function (pdf) of the vector $\boldsymbol{g}$ can be modelled as a Gaussian mixture.

$$p(\boldsymbol{g}|\boldsymbol{z}) = \sum_{k=1}^{K} w_k p(\boldsymbol{g}|z_k)$$

Here K is the number of clusters defined in section 3.3.2.1, and remains an open parameter in our model. The weights w_k stand for the prior probabilities of the components in the mixture and are derived from the results of the clustering stage, by computing the proportion of training images assigned to each cluster. In the Gaussian formulation, the likelihood of the global features given the component is defined as:

$$p(\boldsymbol{g}|z_k) = N(\boldsymbol{g};\, \mu_k^z, \Sigma_k^z)$$

with mean vector μ_k^z and covariance matrix Σ_k^z. Both parameters are obtained from the results of the clustering stage, by computing the parameters of the Gaussian distribution over the set of samples assigned to each cluster (global configuration).

After introducing our simplified model for global features, let us extend it by considering the distributions that depend on local features. For each elementary arms model z_k, we introduce the pdf of each hand $p(h|z_k)$, where h is an index variable with two possible values $h = \{0,1\}$ for the left and right hands respectively. Once the arms model z_k is given, the local centre

coordinates of the selected hand h are also probabilistically modelled by the distribution $p(\boldsymbol{c}|h, z_k)$.
Finally, the likelihood of a point $\boldsymbol{x}$ belonging to the area of interest is expressed by the conditional distribution $p(\boldsymbol{x}|h, \boldsymbol{c}, z_k)$. This distribution models the probability of a pixel to belong to the object being manipulated given the current geometric configuration of arms and hands. It is easy to note that the relative object location and pose is different for various global configurations such as the ones shown in Figure 3-9.

Putting everything together, we can define the partial the model involving the *local features*:

$$p(\boldsymbol{x}|h, \boldsymbol{c}, z_k) = p(h|z_k)p(\boldsymbol{c}|h, z_k)p(\boldsymbol{x}|h, \boldsymbol{c}, z_k)$$

Next, we can define the selected distributions for the local variables as:

- The pdf $p(h|z_k)$ is given by an experimental discrete distribution ($p(h = j|z_k), j = 0,1$)
- The hand centre $\boldsymbol{c}$ follows a Gaussian distribution $p(\boldsymbol{c}|h = j, z_k) = N(\boldsymbol{c};\ \mu_k^c, \Sigma_k^c)$
- The experimental pdf $p(\boldsymbol{x}|h = j, \boldsymbol{c}, z_k)$ is computed also on training set by superimposing all left and right hands from images belonging to the cluster z_k.

The first two pdfs are simply learned by computing their parameters using samples on the training set (see Section 3.3.2.1 for the details). For the third distribution $p(\boldsymbol{x}|h = j, \boldsymbol{c}, z_k)$, it becomes necessary to first crop images by selecting a square region around the hand centre, and then superimpose and accumulate all images belonging to the same global component. The resulting accumulated map for each hand and global configuration is then normalized to sum to one over spatial locations, so as to become a pdf.

In order to compute the saliency map of a particular video frame, the learned distribution is shifted to the hand centre in the frame. Figure 3-13 shows different examples of these distributions for left and right hands, and given five global appearance models. Finally, integrating the distributions of global and local features, the *salience value of a pixe*l $\boldsymbol{x}$ is defined as its likelihood over the proposed model for saliency:

$$S(x) = p(\boldsymbol{x}, \boldsymbol{g})$$

$$S(x) = \sum_{k=1}^{K} w_k p(\boldsymbol{g}|z_k)\, p(\boldsymbol{x}|z_k)$$

$$S(x) = \sum_{k=1}^{K} w_k p(\boldsymbol{g}|z_k) \sum_{j=0}^{1} p(h = j|z_k)p(\boldsymbol{c}|h = j, z_k)p(\boldsymbol{x}|h = j, \boldsymbol{c}, z_k)$$

Let us note that the model in the latter equation allows us to compute the saliency even in the case where one of the arms is absent by simply considering the corresponding probabilities $p(h = 0|z_k)$ or $p(h = 1|z_k)$ as zero.

To summarize, we have developed a probabilistic model that explains how salient pixels are chosen based on hands/arms configuration and the relative expected location of the object being manipulated within each geometric arrangement.

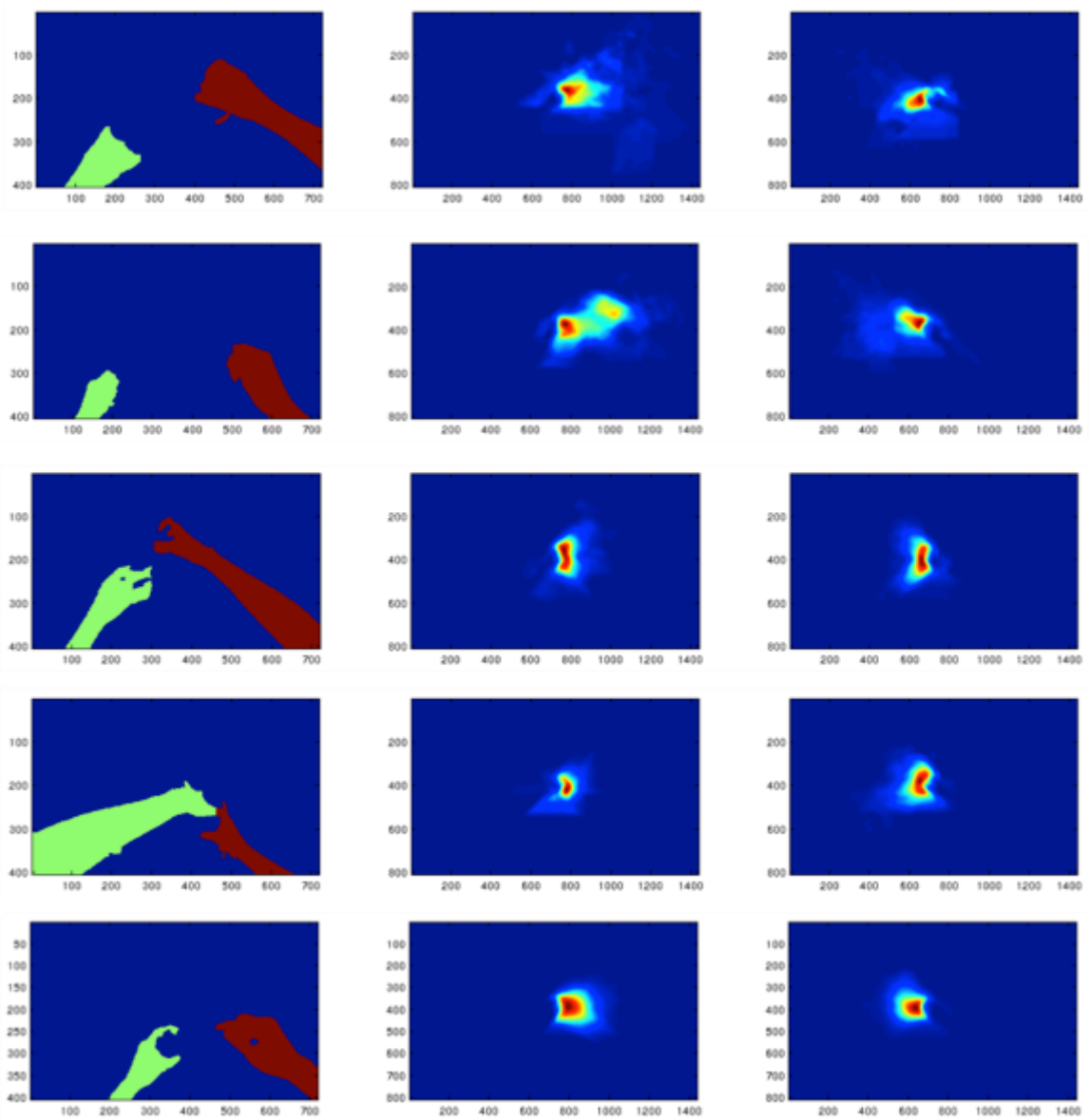


Figure 3-13 Five examples of the obtained experimental distributions $p(\mathbf{x}|h = j, \mathbf{c}, z_k)$. Left column: arm segmentation closest to cluster, Middle column: left hand distribution, Right column: right hand distribution.

3.2.3 Experiments and results

We present the dataset and provide a whole description of the different experimental set-ups for the comparison of our probabilistic top-down saliency model against other saliency approaches. We also assess its contribution into manipulated object recognition performances.

Dataset description

The GTEA dataset we work on was introduced in [3.18]. It is a publicly available database of egocentric videos of 4 subjects performing 7 types of instrumental activities of daily living. The segmentations of arms and objects of interest are provided for 17 videos, a subset of which we use for training our distributions. The frames were annotated with the objects of interest but we manually extend this annotation by drawing bounding boxes on them. The bounding boxes provide the "ground truth" results that could be reached with an "ideal" rectangular salient area. We have split the dataset into training and test sets of videos in such a manner as to even the number of samples of each category in both sets. Let us note that this set-up differs from and is more challenging than the original one proposed in [3.18], where the authors used videos from 3 subjects to train their system and the last one for evaluation.

Selected saliency models for comparison

The following saliency prediction models were selected for comparison due to their popularity or specific relation with the egocentric video.

- The well-known reference model developed by Itti [3.20]. We will denote it as "ITTI" in the follow up of the paper.

- The graph-based visual saliency model developed by Harel [3.21]. It will now be referred to by the acronym "GBVS".

- The spatio-temporal-geometric model presented in [3.22] since it has been specifically developed for saliency extraction in egocentric videos and presents the state-of-the art in saliency-based object recognition in this content [3.2]. This model will be referred as "STG".

- Visual Attention maps built on gaze fixations by reference Wooding's method [3.23]: the fovea projection for each fixation is modelled with a Gaussian of two visual degrees spread and resulting multi-Gaussian surface is normalized.

Figure 3-14 contains computed saliency maps for a randomly selected frame (a). We also display the manually annotated bounding box of the manipulated object (b), as well as the automatically extracted segmentation mask (c)

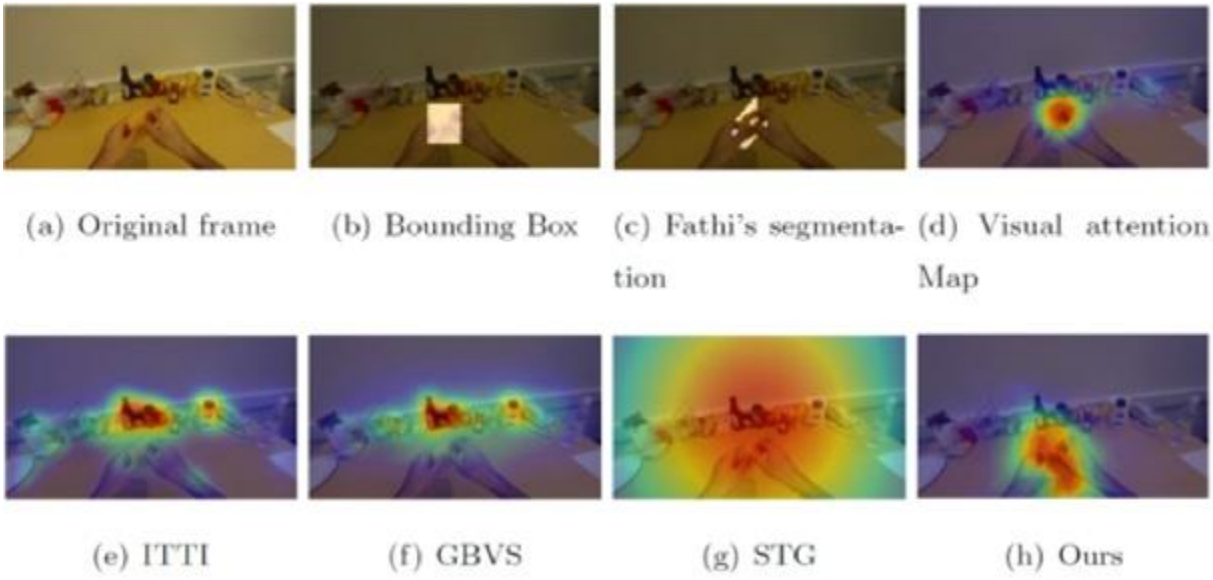


Figure 3-14 Saliency models selected for comparison.

Influence of the number of clusters in the global appearance model

The number of clusters K introduced in Section 3.3.2.1 is an open parameter in our model. We have performed an optimization of the target mean Average Precision (mAP) of object recognition in regard to this parameter using the paradigm introduced in Section 3.3.3.5 Table 3.1 below illustrates the influence of the number of clusters K on the target mAP. Having too few clusters might lead to a lack of information about certain arm models while having too many leads to poorly populated clusters. We found an optimum value at $K = 50$. For the rest of the experiments the saliency model referred as "Ours" corresponds to the methodology presented in Section 3.3.2.3 with K=50 clusters.

Table 3.1 Validation of the number of global appearance models K

	K=20	K=50	K=100
mAP	0.306	0.353	0.342

Psycho-visual evaluation of proposed saliency model

In this section we assess the capacity of our top-down model to predict human visual attention in the task - a guided psycho-visual experiment. The saliency models presented in Section 3.3.3.2 were also assessed for the sake of comparison.

The psycho-visual experiment was designed for recording gaze fixations of subjects who observed the egocentric video with the task of recognition of manipulated objects. For this experiment 31 participants have been gathered, 10 women and 21 men. They were instructed to look at a manipulated object in videos. Each video was watched by 15 subjects or more.

Gaze positions have been recorded with a HS-VET 250Hz Cambridge Research Systems Ltd eye-tracker. The experiment conditions and the experiment room were compliant with the recommendation ITU-R BT.500-11 [3.24]. Videos were displayed on a 23 inch LCD monitor with a native resolution of 960×540 pixels. To avoid image distortions, videos were not resized to screen resolution, but instead a grey frame was inserted around the displayed video. In order to avoid the visual fatigue, the duration of observation was not longer than 15 minutes for each subject. Automatically predicted saliency maps can be compared to human gaze fixation with the help of dedicated metrics. From [3.26] and previous work [3.27], we retained the Normalized Scan Path (NSS) as the most frequently used and suitable for the comparison of saliency maps with human eye fixations:

$$NSS(p) = \frac{SM(p) - \mu_{SM}}{\sigma_{SM}}$$

where p is the location of one fixation and SM is the saliency map with its mean μ_{SM} and standard deviation σ_{SM}. The final NSS score is given by the average of the $NSS(p)$ values for all N eye fixations.

We measured the similarity of recorded eye fixations from the experiment with automatically generated saliency maps from our top-down probabilistic model and the ones presented in section 3.3.3.2. In total 8244 frames were compared for each saliency model and the final mean scores with standard deviations are presented in Table 3.2. Proposed top-down probabilistic model corresponds better to real human eye fixations than the other state-of-the-art saliency models. Since the standard deviation are high, we computed the p-values to back up the hypothesis that the NSS mean using our top down approach is significantly higher than with the other saliency models. At the 1% significance level, the data do provide sufficient evidence to conclude that the mean NSS score using our top-down saliency is greater than the mean obtained using other saliency models.

It is however important to underline that the GVBS and ITTI models are bottom-up and were not designed for a task of recognition of specific objects of interest.

Table 3.2 NSS mean scores (with standard deviations) between human points and different saliency map models.

	ITTI	GBVS	STG	OURS
Mean NSS score	1.05 ± 0.7269	1.29 ± 0.6551	1.52 ± 0.249	$\mathbf{2.28} \pm 1.2226$

Object recognition performances

The ultimate goal of developing a model of top-down visual saliency is in the task of manipulated object recognition. Hence, we first present the object recognition approach with saliency-based psycho-visual weighting of features. This approach, combined with the proposed saliency model, is then compared to other state of the art paradigms for object recognition. We also benchmark it with other saliency models presented in section 3.3.3.2

Saliency-based object recognition approach

In this study we used the saliency-based object recognition method presented in [3.2]. That approach is based on the well-known Bag-of-Visual Words (BoVW) paradigm [3.27, 3.28]. It uses dense SURF descriptors [3.29] and the BoVW is built when weighting each quantized features the estimation of the underlying predicted saliency value. Once each image is represented by its weighted histogram of visual words, an SVM classifier [3.30] is used with a χ^2 kernel. Posterior probabilistic estimates for the occurrence of the object of class C in the frame t are obtained using Platt's approximation [3.31]. For the sake of comparison, we also define our baseline model as the one without any saliency maps. This is a conventional BoVW approach with dense sampling of features on the whole frame. The latter will be referred as "Simple BoVW" in the rest of the paper. For the computation of BoVWs, we use a dictionary size of 4000 visual words. Finally, we define our "ground truth" model as the one where descriptors were extracted only in manually annotated bounding boxes. We consider these bounding boxes as "ideal" saliency maps, referred to as "BoVW with BB".

Comparing with other object recognition approaches and saliency models

Our method outperforms two famous paradigms for object recognition:

- the base-line ("simple") BoVW
- the DPM model [3.32]. It achieves absolute improvements of 10.7% over BoVW and 8.6% over DPM.

Figure 3-15 illustrates the results. Here the "ideal" BoVW with BB is added for the upper bound estimate. As can be seen from the mAP score (last set of bars), our method outperforms the others for this kind of video content, and achieves performance close to the "ideal" case.

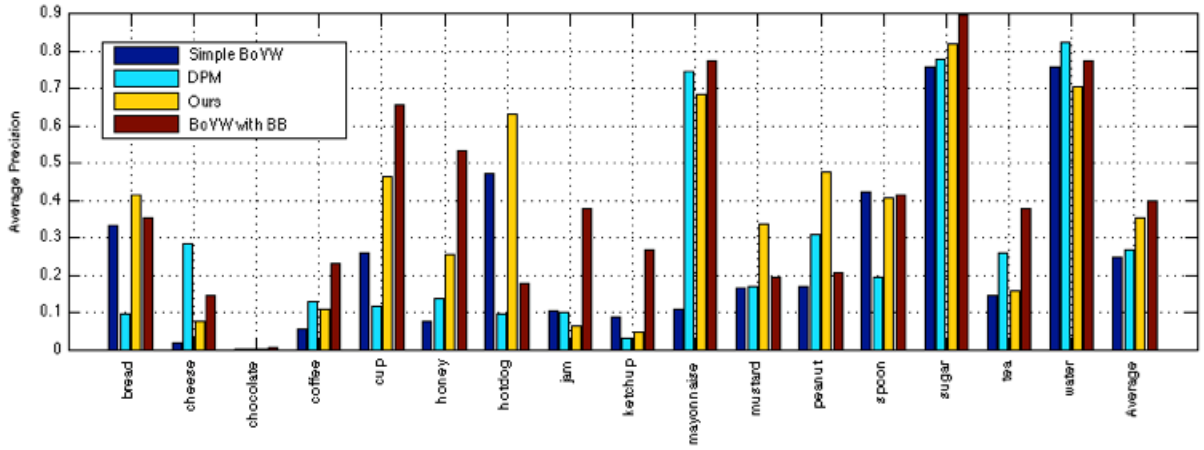


Figure 3-15 Object recognition performances between different paradigms. The results are given in average precision per category and averaged.

In their paper, Fathi et al. [3.18] use a different object recognition method based on the segmented zones. We also computed object recognition accuracy in our test set in the same way it was computed by Fathi, i.e., measuring how well the three highest score detections in each frame match ground-truth labelled objects. The average precision we obtained was slightly higher than Fathi's ,but the comparison is unfair since, as we already mentioned, we have evaluated our detectors under a more challenging set-up with less training data.

We also compare our model with those described in section 3.3.3.2 using the same object recognition approach. Results for per-category and averaged object recognition are displayed in Figure *3-16* in terms of AP.

Compared to the ITTI and GBVS models, our model performs better for almost all categories. These bottom-up saliency models are stimuli-driven, make use of spatial contrast and were not designed to model a top-down, intentional attention component.

The performances of bottom-up STG saliency maps, developed for video were also beaten for almost all categories. This is due to the overestimation by STG of the spread of Gaussian expressing central bias hypothesis on visual attention.

It also achieves slightly better performances than the ones provided by Human Visual Attention maps [3.23]. It is indeed better for some categories since as illustrated in Figure 3-14(d), the visual attention maps are perfectly located but sometimes do not cover the objects of interest enough, contrarily to our model, see Figure 3-14(h) for an example. Student's t-tests with significance level of 0.05 were used to verify the improvements.

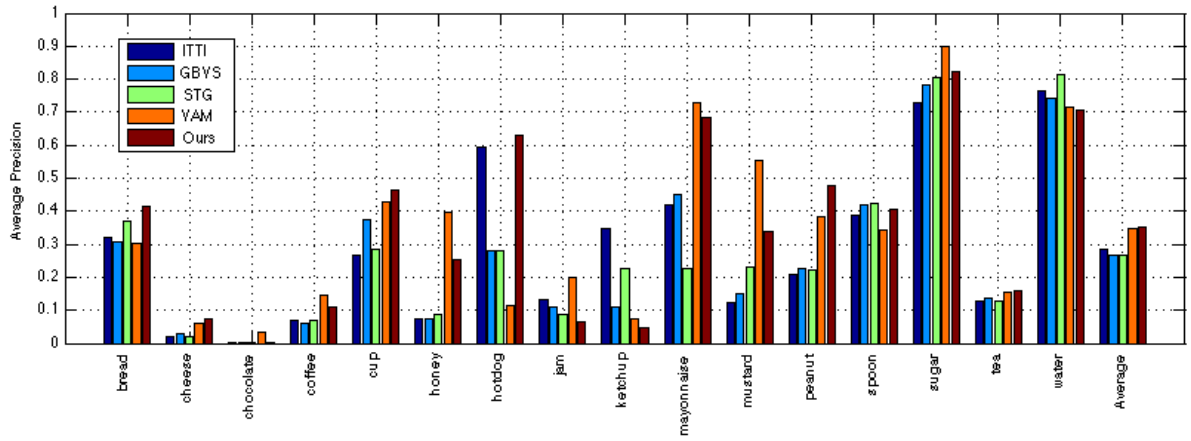


Figure 3-16 Object recognition performances between different saliency models applied to the saliency weighted BoVW paradigm. The results are given in AP per category and averaged.

3.2.4 Conclusions

In the continuation of the work introduced in previous deliverables, we have proposed a new top-down probabilistic visual saliency model for the task of recognizing manipulated objects in egocentric video. It is based on global and local features and uses domain knowledge, i.e. the fact that hands interact with the object of interest. The model predicts well human attention in a task-driven psycho-visual experiment and shows better performances than several bottom-up models widely used in literature, both in terms of comparison with human gaze fixations and target performance in the manipulated object recognition task. Although the model has been developed for the specific case of egocentric video content and the task of manipulated object recognition, the idea behind the method is generic. In top-down visual attention modelling we need to use domain knowledge, contextual information to predict human visual attention. The latter is a complex combination of bottom-up, stimuli driven, and top-down, components. In the perspective of the present research, the combination of bottom-up and top-down prediction and spatio-temporal evolution of visual saliency in a video scene is envisaged with a target application to object and action recognition.

Future plans include obtaining models for the Dem@Care CHUN and DCU datasets and implementing a full study of the method's performance. Later on, this new model will be implemented in the final prototype. We are also studying a way to enhance performances by optimizing the parameters of the model using expectation/maximization.

3.3 References

[3.1] H. Pirsiavash, D. Ramanan, Detecting activities of daily living in first-person camera views, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012, IEEE, 2012.

[3.2] I. Gonzalez Diaz, V. Buso, J. Benois-Pineau, G. Bourmaud, R. Megret, Modeling instrumental activities of daily living in egocentric vision as sequences of active objects and context for alzheimer disease research, in: Proceedings of the 1st ACM International Workshop on Multimedia Indexing and Information Retrieval for Healthcare, MIIRH '13, ACM, 2013

[3.3] Y. Pinto, A. R. van der Leij, I. G. Sligte, V. A. F. Lamme, H. S.Scholte, Bottom-up and top-down attention are independent, Journal of Vision 13 (3).

[3.4] M. Carrasco, Visual attention: The past 25 years, Vision Research 51 (13) (2011) 1484-1525. doi:10.1016/j.visres.2011.04.012.

[3.5] L. Melloni, S. V. Leeuwen, A. Alink, N. G. Muumlller, Interaction between bottom-up saliency and top-down control: How saliency maps are created in the human brain. Cereb Cortex.

[3.6] A. Borj, L. Itti, State-of-the-art in visual attention modeling, IEEE Trans. on Pattern Analysis and Machine Intelligence 35 (1) (2013) 185-207.

[3.7] Y. F. Ma, X. S. Hua, L. Lu, H. Zhang, A generic framework of user attention model and its application in video summarization. IEEE Transactions on Multimedia 7 (5) (2005) 907-919.

[3.8] M. Cerf, J. Harel, W. Einhauser, C. Koch, Predicting human gaze using low-level saliency combined with face detection. in: J. C. Platt, D. Koller, Y. Singer, S. T. Roweis (Eds.), NIPS, Curran Associates, Inc., 2007.

[3.9] T. Huawei, F. Yuming, Z. Yao, L. Weisi, N. Rongrong, Z. Zhenfeng, Salient region detection by fusion bottom-up and top-down features extracted from a single image, IEEE Transactions on Image processing 23 (10) (2014) 4389-4398. doi:10.1109/TIP.2014.2350914.

[3.10] D. Gao, S. Han, N. Vasconcelos, Discriminant saliency, the detection of suspicious coincidences, and applications to visual recognition., IEEE Trans. Pattern Anal. Mach. Intell. 31 (6) (2009) 989-1005.

[3.11] C. Kanan, M. H. Tong, L. Zhang, G. W. Cottrell, Sun: Top-down saliency using natural statistics (2009).

[3.12] A. Torralba, M. S. Castelhano, A. Oliva, J. M. Henderson, Contextual guidance of eye movements and attention in real-world scenes: the role of global features in object search, Psychological Review 113 (2006) 2006.

[3.13] L. Itti, C. Koch, Computational modelling of visual attention, Nature Reviews Neuroscience 2 (3) (2001) 194-203.

[3.14] J. Li, Y. Tian, T. Huang, W. Gao, Probabilistic multi-task learning for visual saliency estimation in video, Int. J. Comput. Vision 90 (2) (2010) 150-165.

[3.15] D. Erhan, C. Szegedy, A. Toshev, D. Anguelov, Scalable object detection using deep neural networks, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.

[3.16] C. Shen, Q. Zhao, Learning to predict eye _xations for semantic contents using multi-layer sparse network, Neurocomputing 138 (2014) 61-68.

[3.17] M. Jones, , M. J. Jones, J. M. Rehg, Statistical color models with application to skin detection, in: Computer Vision and Pattern Recognition, CVPR'1999, IEEE Computer Society, 1999, pp. 274-280.

[3.18] A. Fathi, X. Ren, J. M. Rehg, Learning to recognize objects in egocentric activities, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, IEEE, 2011, pp. 3281-3288.

[3.19] L. S. Al, Z. Shaaban, Normalization as a preprocessing engine for data mining and the approach of preference matrix, in: Proceedings of the International Conference on

Dependability of Computer Systems, DEPCOSRELCOMEX '06, IEEE Computer Society, 2006, pp. 207-214.

[3.20] L. Itti, C. Koch, E. Niebur, A model of saliency-based visual attention for rapid scene analysis, IEEE Trans. Pattern Anal. Mach. Intell. 20 (11) (1998) 1254-1259.

[3.21] J. Harel, C. Koch, P. Perona, Graph-based visual saliency, in: ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS 19, MIT Press, 2007, pp. 545-552.

[3.22] H. Boujut, J. Benois-Pineau, R. Megret, Fusion of multiple visual cues for visual saliency extraction from wearable camera settings with strong motion, in: A. Fusiello, V. Murino, R. Cucchiara (Eds.), Computer Vision (ECCV 2012) Workshops and Demonstrations, Vol. 7585 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2012, pp. 436-445.

[3.23] D. Wooding, Eye movements of large populations: II. Deriving regions of interest, coverage, and similarity using fixation maps, Behavior Research Methods 34 (2002) 518-528, 10.3758/BF03195481.

[3.24] International Telecommunication Union, Methodology for the subjective assessment of the quality of television pictures, Recommendation BT.500- 11, International Telecommunication Union (2002).

[3.25] N. Riche, M. Duvinage, M. Mancas, B. Gosselin, T. Dutoit, Saliency and human fixations: State-of-the-art and study of comparison metrics, in: The IEEE International Conference on Computer Vision (ICCV), 2013.

[3.26] O. L. Meur, T. Baccino, Methods for comparing scanpaths and saliency maps: strengths and weaknesses. Behavior Research Methods.

[3.27] J. Sivic, A. Zisserman, Video Google : A text retrieval approach to object matching in videos, in: Proceedings of the International Conference on Computer Vision, Vol. 2, 2003, pp. 1470-1477.

[3.28] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, C. Bray, Visual categorization with bags of keypoints. In Workshop on Statistical Learning in Computer Vision, ECCV, 2004, pp. 1-22.

[3.29] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, Speeded-up robust features (SURF), Comput. Vis. Image Underst. 110 (3) (2008) 346-359.

[3.30] C. Cortes, V. Vapnik, Support-vector networks, Machine Learning 20 (1995) 273-297.

[3.31] J. C. Platt, Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods, in: Advances in Large Margin Classifiers, MIT Press, 1999, pp. 61-74.

[3.32] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (9) (2010) 1627-1645.

4 Activity Monitoring

4.1 Introduction

The work on activity monitoring has continued with improvements on the methods being developed for activity localization and recognition. The ultimate goal is to achieve accurate recognition of activities of daily living, so that visual monitoring can take place autonomously, relieving the formal and informal caregivers of a large part of their daily obligations. Activity localization methods have been developed in order to isolate activities of interest in videos of long duration, such as those recorded during daily life monitoring. Subsequent recognition was shown to be accurate, demonstrating the correct spatiotemporal localization of activities.

Figure 4-1 Dem@Care lab experiments: From left to right and top to bottom Dem@Care1: Eat Snack, Enter Room, HandShake, Read Paper, Dem@Care2: Serve Beverage, Start Phonecall, Drink Beverage and HandShake, Dem@Care3: Prepare Drug Box, Prepare Drink, Turn On Radio, Water Plant. Dem@Care4: Answer phone, Prepare Drug Box, Prepare Hot Tea, Establish Account Balance.

Figure 4-2 Dem@Care home experiments: From left to right and top to bottom Dem@Home1:Wash Dishes, Prepare Meal, Eat. Dem@Home2:Sit on couch, Open fridge, kitchen activity.

This work was further refined in several manners: (1) new real world data was collected in the lab environment in the premises of the Greek Association for Alzheimer's and Related Disorders (GAADRD) and two smart home environments that setup on real MCI patients, as we can see in Figure *4-1* and Figure *4-2* respectively, on which experiments took place for activity recognition accuracy.

The activity localization and recognition methods developed within Dem@Care were tested on this data. (2) In parallel, new methods were developed to increase the speed of activity recognition using RGB and Depth video frame, ultimately aiming at a real time detection and recognition system. (3) In order to generalize the applicability of the proposed approach, a novel method for handling multiple camera motion planes is introduced, leading to better activity recognition results due to the improved camera motion compensation.

4.1.1 Objectives

The objectives of our work have been to further expand our activity localization and recognition framework, to be able to deal with more challenging videos. The goal is to provide a method that can operate more quickly (eventually aiming at near real time performance), to examine its operation on new benchmark videos in real world conditions and to expand our approach to be able to work with moving camera data. More specifically, in this deliverable we present:

- Improvement and expansion of the camera motion compensation method for spatial localization and activity recognition.
- Speeded up activity recognition using computationally efficient motion estimation.
- Collection of new RGB-D data in the Dem@Lab and Dem@Home installation for activity localization and recognition in real world conditions.

4.1.2 Description of the method

Camera Motion Compensation for Improved Activity Recognition

The superpixels-based [4.1] method, previously described in D4.4 and presented in [4.8], which produced multiple homographies corresponding to the camera motion at multiple depths, is further improved and tested on benchmark datasets to demonstrate its efficacy and usefulness. Each video frame is segmented into superpixels, which are used to estimate a global homography between two image frames. Bad matches are eliminated from this homography via the application of RANSAC. Local homographies are then estimated between tracked interest points in frame *t* and their re-projections in frame *t+1*, using the previously estimated global homography. The resulting local homographies are employed for motion compensation, leading to "motion planes", containing pixels with the same local camera motion but a different appearance (as they correspond to different superpixels).

Experiments with benchmark datasets, such as the challenging UCF videos, containing real-world recordings of sports events with a moving camera, demonstrate the effectiveness of this method. The figure below contains several sample frames where the moving person is detected accurately, despite the camera motion. It should be noted that the UCF dataset contains ground truth, depicted in the frames by the green rectangle, so we can quantify the accuracy of our method. Indeed, the mean Intersection over Union (IoU) for this dataset is 56.2%, compared to 39.9% for the current state of the art [4.2].

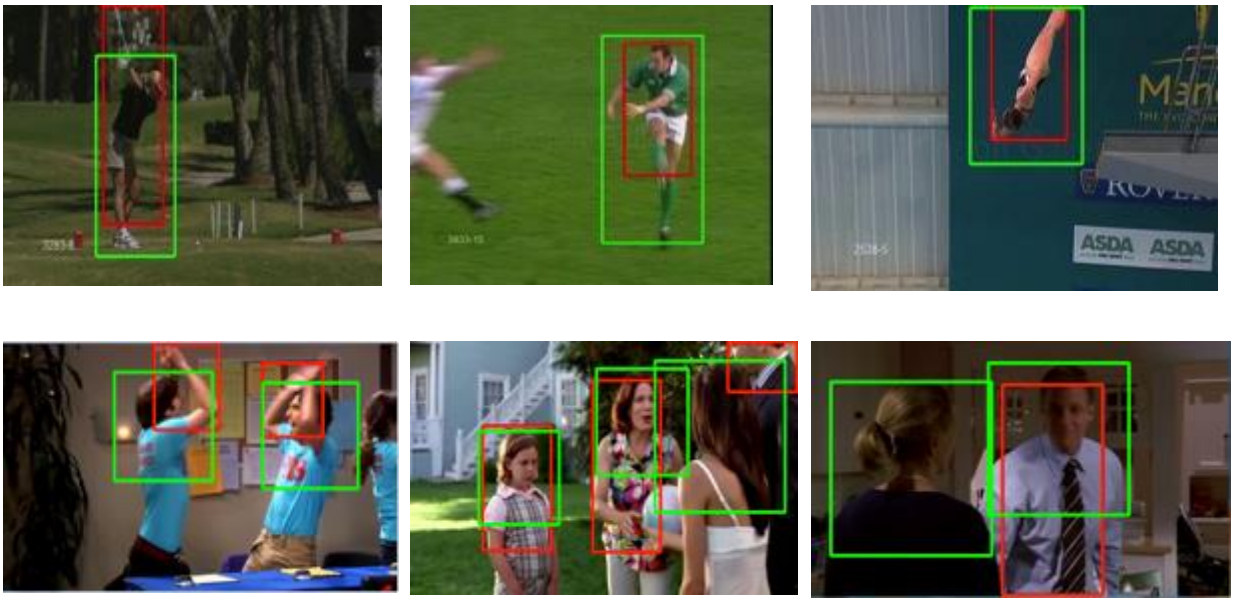


Figure 4-3 Moving person detection in videos recorded with a moving camera. The green boxes denote the ground truth and the red boxes the detected people using our multiple homographies approach.

Speeded Up Activity Recognition

As the recognition of activities of daily living, and activities in general, is needed in more and more applications, we propose an improvement upon our existing activity recognition scheme by replacing the computationally costly optical flow estimation with simpler, but faster, block matching. This work was applied in several activities of daily living action datasets, including Dem@Care recordings, presented in [4.9] and extended in [4.10]. Block matching is applied to the video, leading to a sparse motion field: in order to achieve the most accurate motion estimation results, we apply Full Search block matching, however we also apply a termination criterion using a threshold as in PMVFAST. This leads to accurate motion estimates, with a lower computational burden than full search. At the same time, the motion estimate avoids getting trapped in local minima, due to the use of the termination criterion. The resulting motion vectors are used to estimate Motion Bondary Activity Areas (MBAAs) and sample dense interest points in them, around which multi-scale HOG and HOF descriptors are estimated. The resulting feature vectors are described by a GMM model, followed by Fisher encoding, to be incorporated in an SVM-based recognition scheme.

Experiments took place with benchmark datasets, namely the University of Rochester Activities of Daily Living (URADL) videos, as well as video data collected for the purposes of the Dem@Care project in a lab environment, at CHUN in Nice, France, and at GAADRD in Thessaloniki, Greece. We compared the results of using block-matching for motion estimation with the results using variants of optical flow as in the SoA, within a SoA activity recognition framework. Table 4.1 below shows that the proposed method indeed succeeds in achieving SoA accuracy at a lower computational cost. Tests took place with the video data at several resolutions, for an in-depth examination of the effect of resolution on the block matching motion estimation accuracy and the corresponding activity recognition accuracy. For reasons of space, we only present results for a resolution of 640x480 in this work.

Table 4.1 Activity Recognition Accuracy for 640x480 resolution, with block matching speedup

Motion estimation	**URADL (%)**		**CHUN (%)**		**Dem@Care1 (%)**	
	Accuracy	Speedup	Accuracy	Speedup	Accuracy	speedup
Dual TV-L1 [4.3]	**90.0**	3x(original video was 1280x720)	98.8	1x	83.3	1x
VarFlow [4.4]	88.0	9.8x	95.4	2.2x	82.1	1.6x
Block Matching (FS)	88.7	38.0x	**97.3**	**6.5x**	79.9	4.1x
Block Matching (DS)	**88.7**	**39.2x**	97.0	13.4x	**81.8**	**4.3x**

Appearance and Depth for Rapid Human Activity Recognition in Real Applications

A novel technique for activity localization and recognition from color-depth sequences recorded with the Kinect sensor, specifically tailored for the recognition of Activities of Daily Living (ADLs), was presented in [4.7]. Comparative analysis with SoA [4.5, 4.6] on three challenging ADL datasets indicates that our algorithm is very appropriate for real life scenarios as it achieves SoA accuracy while performing 10-20 times faster.

Our RGB-D video processing framework consists of four stages (Fig.4-2): i) the depth image is refined in order to fill the missing values produced by the sensor. ii) Activity is detected on a grid of Spatio-Temporal Activity Cells (STACs) applied throughout the video sequence. iii) Activity Representation takes place by extracting features from the 3D volume comprised of all STACs that contain activity, iv) the features are encoded with Fisher Vectors of fixed size and Activity Recognition is implemented with a SVM classifier.

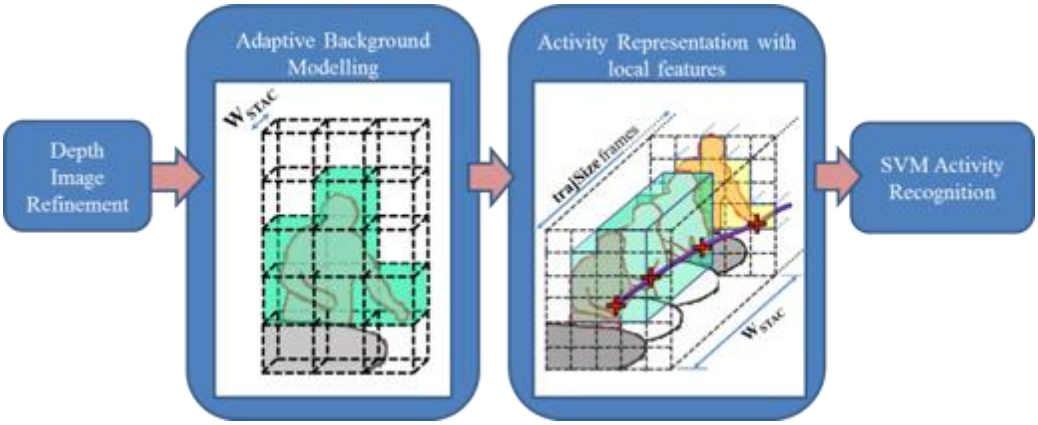


Figure 4-2. Overview of our Activity Localization and Recognition solution: (from left to right) a) depth frame refinement corrects noisy depth values. b) Adaptive background modelling uses HOG and HoD to separate "active" from "inactive" STACs. c) HOG, HOSNP and 3D trajectories are accumulated over time to represent human activities. d) Fisher encoding over the whole video trains a multiclass SVM model.

Experiments took place on three Dem@Care datasets (i.e. Dem@Care1, Dem@Care2, Dem@Care3) of elderly people performing ADLs, available for benchmark purposes upon request. The 640x480 videos of these datasets contain a variety of activities (e.g. Drink Beverage, Eat Snack, Talk to Visitor, Start Phonecall, End Phonecall, Prepare Hot Tea, Read Article, etc) and a great deal of anthropometric differences between subjects. Moreover, each dataset is recorded in a different environment at a unique sampling rate.

Results demonstrate that our method achieves accuracy that is highly competitive to SoA algorithms [4.5, 4.6] that make use of Optical Flow, while maintaining a very low computational cost. More specifically, the proposed method resulted in a -0.6% accuracy compared to SoA for D1 while performing 14.8 times faster. Similarly, a -3.3% accuracy deficit from SoA was reported for D2 with a 11.8 faster computation, and lastly, our algorithm outperformed SoA on D3 (+1.2% accuracy) while performing 21.4 times faster.

Further analysis exposes the value of the descriptors chosen for activity representation. Activity Recognition was carried out with different combinations of descriptors and as shown in Table 4.2. The combination of HOG and HOSNP significantly increases the mean accuracy, demonstrating that these descriptors incorporate different aspects of the video data. Lastly, the concatenation of 3D trajectories boosts the accuracy even further, as more motion information is introduced to the final descriptor.

Table 4.2: Average Accuracy of the proposed method for different combinations of descriptors and average accuracy of SoA methods[1,2]

Dataset		HOG	HOSNP	HOG+HOSNP	HOG+HOSNP+3DTraj	[1]	[2]
Dem@Care1	Av. Acc	70.2%	81.4%	85.3%	**89.6%**	85.1%	90.2%
	Speed up	**x14.8**				x2.2	x1
Dem@Care2	Av. Acc	66.9%	69.9%	75.0%	**79.9%**	83.2%	79.9%
	Speed up	**x11.8**				x2.3	x1
Dem@Care3	Av. Acc	80.7%	79.1%	88.6%	**94.5%**	93.3%	91.7%
	Speed up	**x21.4**				x3.9	x1

Activity Detection and Isolated Activity Recognition for new GAADRD Dem@Lab recordings

The activity recognition method described in D4.4 and earlier deliverables is tested on new Dem@Lab recordings that took place in the GAADRD premises in Thessaloniki, Greece. The resulting activity recognition results are referred to as "isolated activity recognition" as each video segment contains only one of the activities of interest. The performance of activity detection was also tested on this dataset and shown to lead to reliable activity recognition results on videos of a long duration. In particular, the new recordings involved 25 patients suffering from MCI (Mild Cognitive Impairment) who were asked to carry out the following activities: answer phone, establish account balance, leave room, prepare drug box, prepare hot tea, read article, turn on radio, water plant. In Table 4.3, Table 4.4, Table 4.5 and Table 4.6, we see that most activities achieve high accuracy rates using the HOGHOF+Traj and Fisher framework to represent them.

Table 4.3 Isolated activity recognition for Dem@Care1

	CU	DB	EP	ER	ES	HS	PS	RP	SB	SP	TV
CU	0,853		0,029		0,059		0,029			0,029	
DB	0,02	0,939			0,041						
EP	0,031		0,844		0,031					0,094	
ER				1							
ES		0,244	0,022		0,711			0,022			
HS						0,906					0,094
PS		0,029			0,086		0,714		0,171		
RP	0,031			0,031				0,938			
SB					0,029		0,059		0,912		
SP			0,061		0,061					0,879	
TV											1
AA	0,881455										

Table 4.4 Isolated activity recognition for Dem@Care3

	AP	EAB	PDB	PHT	RA	TOR	WP
Phone	**96,00%**					4,00%	
Account	8,33%	**75,00%**	16,67%				
Drug			**95,24%**				4,76%
Tea				**92,00%**	4,00%	4,00%	
Read					**95,45%**		4,55%
Radio						**100,00%**	
Plant					10,00%		**90,00%**
AvAcc	**91,96%**						

Table 4.5 Isolated activity recognition for Dem@Care4

	AP	EAB	PDB	PHT
Phone	**98,41%**		15,87%	
Account		**100,00%**		
Drug	16,67%		**93,33%**	5,00%
Tea	16,13%			**98,39%**
AvAcc	**97,53%**			

Table 4.6 Isolated activity recognition for Dem@Home1

	Drink/Eat	PrepareMeal	UseFridge	WashDishes
Drink Eat	**96,30%**	1,20%	1,00%	1,50%
PrepareMeal	5,20%	**93,00%**		1,80%
UseFridge	1,60%		**98,40%**	
WashDishes	8,80%			**91,20%**
AvAcc	**94,70%**			

After activity detection in videos of a long duration, recognition takes place using the Statistical Sequential Boundary Detection (SSBD) method described in D4.4. As we can see from Table 4.7, the detection and recognition of activities in long duration videos leads to lower recognition accuracies, originating from errors in the detection of activity boundaries. Nonetheless, initial results are quite accurate, and encouraging for future work and improvements upon the method.

Table 4.7 Activity detection for the new Dem@Lab and Dem@Home datasets

Datasets	Recall	Precision
Dem@Care1	60.78%	n/a
Dem@Care3	80.89%	44.36%
Dem@Care4	71.47%	55.16%
Dem@Home1	63.25%	2.51%
Dem@Home2	91.33%	3.4%

4.1.3 Discussion and results

We see in the previous sections that the proposed activity recognition method for moving camera leads to accurate results on benchmark datasets. This method is expected to provide more reliable results in the case where multiple homographies can indeed be detected in a scene, i.e., when the scene contains various depths, where the camera motion appears to be different in different regions.

We present a method for speeding up the activity recognition by replacing its most computationally costly component, namely the optical flow estimation, with block matching. Block based matching for motion estimation is quite an old method, which however is shown to lead to accurate results while achieving faster activity recognition. In theory even faster activity recognition could be achieved, via the elimination of false alarm motion estimates, which led to many short length trajectories that could be eliminated.

Activity recognition after detection using the SSBD is shown to lead to accurate results, which can be improved on several levels. Adequate amounts of training data, where the activities of interest are at least partly visible, are required to ensure better accuracy.

4.1.4 Conclusions

We have further refined and expanded our methods for activity detection and recognition. Improved speed is achieved with the use of block-based matching and, as noted above, has room for improvement for even better speedups. Moving camera results tested on benchmark datasets led to accurate spatial localization of actors in a scene, as comparisons with ground truth data showed. Future work includes refinements of the activity descriptor and recognition method, as well as the activity detection approach used, for higher accuracy in long duration videos. Furthermore, options for improving activity detection and recognition without loss in performance will be investigated. Testing will continue on real world data, obtained from recordings for the Dem@Care project.

4.2 References

[4.1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Susstrunk, “SLIC superpixels compared to state-of-the-art superpixel methods,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.

[4.2] Shugao Ma, Jianming Zhang, Nazli Ikizler-Cinbis, and Stan Sclaroff, "Action recognition and localization by hierarchical space-time segments," in *Int'l Conf. on Computer Vision*. IEEE, 2013, pp. 2744–2751.

[4.3] C. Zach, T. Pock, and H. Bischof, "A duality based approach for realtime TV- L1 optical flow," *Pattern Recognition*, pp. 214–223, 2007.

[4.4] A. Bruhn, J. Weickert, C. Feddern, T. Kohlberger, and C. Schnorr, "Variational optical flow computation in real time," *IEEE Trans. on Image Processing*, vol. 14, no. 5, pp. 608–615, 2005.

[4.5] K. Avgerinakis, A. Briassouli, and I. Kompatsiaris. Rexognition of activities of daily living for smart home environments. In 9th International Conference on Intelligent Environments (IE2013), 2013.

[4.6] H. Wang and C. Schmid. Action recognition with improved trajectories. In Computer Vision (ICCV), 2013 IEEE International Conference on, pages 2087-2090, 2012.

[4.7] S. Tachos, K. Avgerinakis, A. Briassouli, I. Kompatsiaris, "Appearance and Depth for Rapid Human Activity Recogntion in Real Applications", British Machine Vision Conference (BMVC), Sep. 2015

[4.8] K. Avgerinakis, K. Adam, A. Briassouli, Y. Kompatsiaris, "Moving camera human activity localization and recognition with motionplanes and multiple homographies", IEEE International Conference on Image Processing (ICIP), Quebec city, Canada, Sep. 2015.

[4.9] S. Poularakis, K. Avgerinakis, A. Briassouli, Y. Kompatsiaris, "Computationally efficient recognition of activities of daily living", IEEE International Conference on Image Processing (ICIP), Quebec city, Canada, Sep. 2015.

[4.10] S. Poularakis, K. Avgerinakis, A. Briassouli and I. Kompatsiaris, "Efficient Recognition and Coding of Activities of Daily Living", to be submitted.

5 Lifelogging

5.1 Introduction

In the context of Dem@Care, the purpose of lifelogging is to facilitate reminiscence therapy and to discover life patterns in longitudinal data. For the clinician and/or the PwD we designed an image browser, where they can review life narratives captured by lifelogging devices. Most applications of lifelogs benefit from automatically structuring them into discrete events. The challenges of effective structuring, searching and browsing of a lifelog in order to locate important or significant information has been addressed as a media process which is based on 1) the capturing and uploading of sensor data, images or video 2) postprocessing of the uploaded data and 3) access to processed data. This has been described in detail in [5.1] which presents the lifelog as a repository from which information – events of importance -- can be retrieved, which has been the access paradigm for the lifelog.

In [5.2], a method that can automatically segment a collection of lifelog images captured from a wearable camera is described. The features used to compare the similarity between images were MPEG-7 descriptors, namely colour layout, colour structure, scalable colour and edge histogram; similarity scores across adjacent images were calculated using those features. The authors used a technique called peak scoring to detect the dissimilarity and some automatic threshold methods were applied to determine the boundaries between discrete events. In the final step of this process, event boundaries that are too close to each other are merged. Following this approach, other researchers apply machine learning techniques such as support vector machines (SVM) to train a classifier so as to identify the boundaries between events in a sequence of lifelog images. External data from other sensor sources such as accelerometers, GPS co-ordinates or metadata, could also be used in the segmentation process.

Once images have been segmented into events, a single image is selected to represent the entire event in order to facilitate event queries from users. Several selection methods have been investigated including selecting the middle image, selecting the image that is most representative, and selecting the image that is most representative but also most different to other events. Image quality was also considered as an important criterion in selecting key frame images and different image quality measures have been evaluated.

When a lifelog is segmented into events for event-based access, by default we get date and time, and perhaps location, as keys by which we can access those events. However, we also need to analyse the lifelog content itself and leverage the rich information it contains. A standard approach to multimedia access is to build a set of classifiers for a set of pre-defined semantic concepts and to train each classifier so that it assigns images from the lifelog, and a score for the confidence of that semantic concept's presence in the image. In [5.3] thresholds were applied to determine whether a lifelog image belongs to a concept or not. One of the most important statistics for concept detection is the author-calculated average number of concepts detected for each event and compared among users.

While indexing lifelog events by the presence or absence of a set of concepts is useful, [5.4] described a way that a user can retrieve events by using queries which are far more semantically relevant and which can encapsulate different aspects of an information need,

specifically the when, where, who, what aspects. This also allows for similar events to be retrieved by computing and ranking the similarity between events. Other lifelogging research [5.5] has shown an interest in building ontology of semantic concepts that occur in everyday activities and which can be detected in lifelogging image collections. Wang [5.6] used Markov chains to model the probability distribution of objects and of semantic concepts detected in lifelog image events.

Despite all the research carried out into applications of lifelogging and into post-processing of lifelog data, especially visual lifelogs consisting of images from wearable cameras, research concentrating on analysis of lifelogs which investigates longitudinal aspects and the causality and impact of patterns detected from longitudinal analysis on lifestyle, is not apparent.

5.2 Pattern Discovering in Lifelog Data

5.2.1 Periodicity Detection

Researchers in lifelogging are just now starting to realise the potential offered by aggregated lifelogs that bring together data from multiple sensors, for a single individual. Current research into lifelogging does not fully exploit temporal relationships when dealing with data [5.7]. In [5.8] time series analysis methods were used to study chronologically presented lifelogging images. The authors concluded that DFA (Detrended Fluctuation Analysis) shows that lifelogging data is not a random walk, but is closer to a time series with a cyclic fluctuation. The work presented in this paper builds upon this finding. Detecting patterns of periodicity would give huge insights and reveal aspects of a person's lifestyle. However, periodicity detection usually relies on data that is both complete and has no missing values, and is accurate with no probabilities associated with the data. With lifelogging, this isn't always the case as people can simply decide not to switch on their logging devices or there can be calibration errors with the lifelog sensors. In this paper we address how to detect repeating patterns of lifestyle from lifelogs when the underlying data has missing or incomplete data, or even data that is erroneous. Once such patterns and periodicities have been detected, it is beyond the scope of this paper to determine how to use them or present them back to users. To illustrate our work on detection from noisy data, we examine real lifelogs, which have in-built gaps and noise. Our work demonstrates that even with very noisy data, which is far from being continuous, we can detect repeating patterns and periodicities.

5.2.2 Periodicity Methodology

Our aim is to detect and report longitudinal patterns in lifelogs, which we can regard as a form of time series, and these patterns can be referred to as periodicities. Signal processing theory tells us that in order to detect low-level periodicities in any time-series, we calculate its power spectral density (PSD) [5.9]. The PSD essentially tells us the strength of the expected signal power at each possible frequency of the signal. Because frequency is the inverse of period, we wish to identify frequencies that carry most of the energy and then from that detect the most dominant periods. Two estimators of the PSD could be used to detect and present periodicities; the periodogram and the circular autocorrelation or full cross correlation. The power spectral density can be computed using the DFT (Discrete Fourier Transform) or FFT (Fast Fourier Transform). PSD is also called periodogram and we can detect and visualise periodicity using a periodogram. The periodogram is visualised as a 2D plot with spectral

frequencies on the x-axis and the strength of the pattern at each frequency measured on the y-axis.

In terms of lifelogging, the periodogram can be used to detect the natural cycles that occur in lifestyle, behaviour, and activities. Periodicity can be observed in many natural phenomena, such as circadian rhythms associated with our sleep, annual seasons and so on. Intuitively, we think of our routine daily lives as composed of various forms of recurring events with obvious periodicities around daily, weekly, monthly, seasonal and annual cycles. In any kind of spectral analysis of a lifelog we expect to see periodicity around these frequencies. However, without the help of lifelogging devices and the resulting lifelog, analysing the periodicity of human life is not a practical proposition.

We now define the tools we use to detect periodicity in lifelogs.

A) Autocorrelation:

In statistics, correlation is basically measuring how similar two sequences are. This quantitative measurement of the similarity between signal 1 and signal 2 can be defined as:

$$r_{12} = \frac{1}{N}\sum_{n=1}^{N-1} x_1[n]x_2[n]$$

Cross correlation between time shifted sequences, can be defined as:

$$r_{12}(k) = \frac{1}{N}\sum_{n=1}^{N-1} x_1[n]x_2[n+k]$$

All possible k-shifted time series could generate another sequence of numbers only changing with k, which is called full cross-correlation. The correlation between a signal and time shifted version of itself is called an auto-correlation. A lag operator is used to generate the time-shifted signal and '0 lag' equals to mean-square signal power. Auto-correlation can be defined as

$$r_{11}(k) = \frac{1}{N}\sum_{n=1}^{N-1} x_1[n]x_1[n+k]$$

B) Periodogram

The normalized Discrete Fourier Transform (DFT) of a sequence x(n), n = 0,1,...,N −1 is a sequence of complex numbers X(f):

$$X(f_{k/N}) = \frac{1}{\sqrt{n}}\sum_{n=0}^{N-1} x(n)e^{-\frac{j2\pi kn}{N}}$$

where the subscript k/N denotes the frequency that each coefficient captures. Suppose that X is the DFT of a sequence

$$P(f_{k/N}) = \|X(f_{k/N})\|^2 \quad k = 0, 1, \ldots, \lceil\frac{N-1}{2}\rceil$$

Notice here that k ranges from 0 to (N-1)/2. In order to find the k dominant periods, we need to pick the k largest values of the periodogram. This works well for short to medium length periods, but performance is worse for long periods or low frequencies, because each value in

the periodogram indicates the power at frequency interval [N/k, N/(k-1)], which is too wide to capture large periodicity. Thus the accuracy of periodicity detection at low frequencies will be lower than at higher frequencies. For lifelogging, this means there is difficulty in detecting patterns measured in years. Another difficulty when using periodograms is spectrum leakage, which causes frequencies that are not integer multiples of the DFT bin width to disperse over the entire spectrum, potentially resulting in false alarms being in the periodogram. Nevertheless, the periodogram is still a good way to guarantee the accuracy of detected periods with short to medium frequency.

In the context of our work on periodicity detection from lifelogs, one of the challenges we are faced with is missing or erroneous data from the lifelog. For such a scenario, the Lomb-Scargle periodogram [5.10] can be used to detect periodicity in signals with missing, unevenly or unequally spaced data. This is defined formally as

$$P_X(\omega) = \frac{1}{2}\left\{\frac{[\sum_{n=1}^{N} y(t_n)\cos(\omega(t_n-\tau))]^2}{\sum_{n=1}^{N}\cos^2(\omega(t_n-\tau))} + \frac{[\sum_{n=1}^{N} y(t_n)\sin(\omega(t_n-\tau))]^2}{\sum_{n=1}^{N}\sin^2(\omega(t_n-\tau))}\right\}$$

where τ is defined as:

$$\tan(2\omega\tau) = \frac{\sum_{n=1}^{N}\sin(2\omega t_n)}{\sum_{n=1}^{N}\cos(2\omega t_n)}$$

The purpose of this work is to determine how well periodicity can be detected in lifelog data, focussing specifically on how the tools perform in the scenario of missing data and gaps in the lifelog.

5.2.3 Intensity of Periods

In order to calculate the degree to which a frequency is periodic, we introduce intensity of periodicity. The ideal output of periodicity intensity is a series of numbers that indicate the regularity of a certain activity's period. The Intensity could potentially reveal changes in periodic data, which in turn may indicate a change of behaviour. Periodicity intensity provides a different and practical way for clinicians to review data generated by PwD.

After identifying the significant periodicity (e.g., weekly, daily) we would like to compute how strong or weak the period is and how the strength of the period changes with time. Temporal and spectral analysis is a popular signal processing approach to understand how the frequency of a signal changes with time. Assuming we are able to detect the most significant periods in lifelog data, it is interesting to see how the strength of the periods changes, so that we can identify when is high/low regularity. An intuitive idea is to calculate energy carried by the most significant periods by using a moving window.

1) Choose suitable length of window, within which the periodogram can be calculated.

If the window size is too long, the temporal resolution of the signal will be poor, while achieving good frequency resolution, and vice versa. One way to balance the trade-off between time and frequency is to use overlapping windows, but what needs to be addressed is

that window overlap brings a time lag into the periodicity's intensity, which may appear as a delay or advance in the intensity graph, compared to real-world data.

2) Extract a frequency that is exactly and/or close to detected significant periods and the corresponding energy from the periodogram.

Depending on size of the window, the most significant period within a window may differ from the most significant period detected using all data. Also the most important periods detected using all data could be affected by spectral leakage, as the real frequencies may not be the integer times of frequencies of cosine/sine basis in FFT. Using sophisticated methods such as adding various window functions could decrease the spectral leakage problem.

3) Moving window and repeating 2nd step, until there are no more datapoints available.

The y-axis in the figure stands for the regularity of the selected frequency. Note that the high total amount of energy within a window might lead to high intensity values.

5.2.4 Dataset (Periodicity)

Sleep Data

The first dataset represents 2.5 years of continuous night sleep monitoring for an individual with a capture rate of more than 80%. Data was collected using the wrist-worn Lark sleep sensor and contains the following information:

1) Time to sleep – represents the time between going to bed and falling asleep;

2) Time to rise – represents the time between waking and getting out of bed;

3) Time asleep – represents the duration of sleep;

4) Quality – a numeric indicator of sleep quality computed as a function of how well the night's sleep mapped to the circadian sleep (90-minute) rhythm and how many cycles of that rhythm were completed;

5) Times woken up – represents the number of instances a person wakes up during sleep, where "wake up" even represents turning over in bed;

The distribution of parameters 3 and 4 is shown in Figure 5-1. An obvious periodicity we would expect to detect is the weekly cycle, where the subject sleeps longer during weekends than workdays because they has a regular work schedule from Monday to Friday.

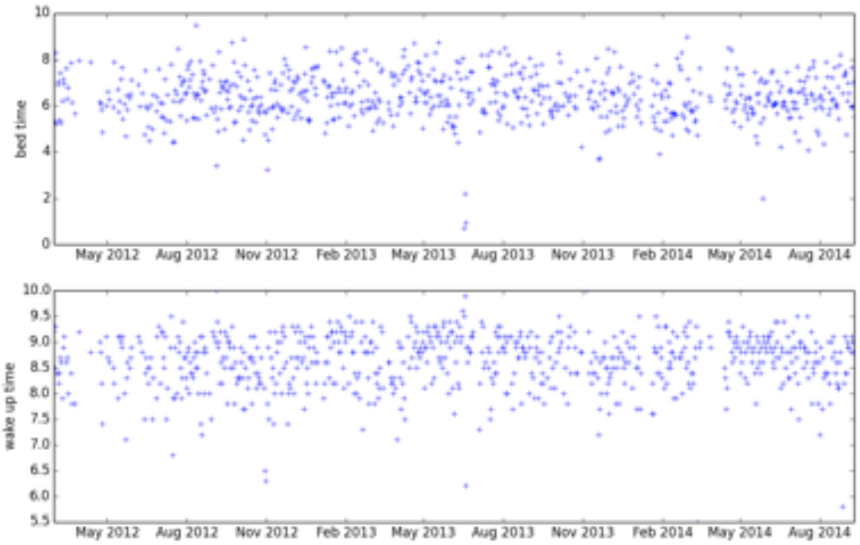


Figure 5-1: Visualisation of raw sleep data

Sports Data

The second dataset represents a 10-year log of physical exercise activities including running, cycling and swimming, from an international tri-athlete (now retired from competition). The log contains a daily entry for the distance covered for 1 or more of the sports, as well as daily text comments which can indicate mood, training effort, relative performance, weather, etc. and these can be analysed for sentiment. Sports datasets capture 100% of activity logs over 10 years. Obvious periodicities to be detected from this data include seasons, performance at targeted sports events, perturbations caused by occasional injury and overall decline over the decade from ageing.

In Figure 5-2, the raw distances for running, cycling, swimming and for aggregated activity effort is shown. The latter of these plots accounts for days where the athlete would exercise or compete in more than one discipline, and their aggregated activity is computed according to the metabolic equivalent (MET) where the unit of MET is 1 kcal/kg*h. To calculate this the average speed for each of the three sports activities of the athlete is used. In [5.11], the MET for each sport activity at the average speeds indicated by the athlete are shown in Table 5.1.

Table 5.1: MET table

Activity	Speed (kph)	MET
Running	13	12.9
Cycling	25	8.4

Swimming	3	8.9

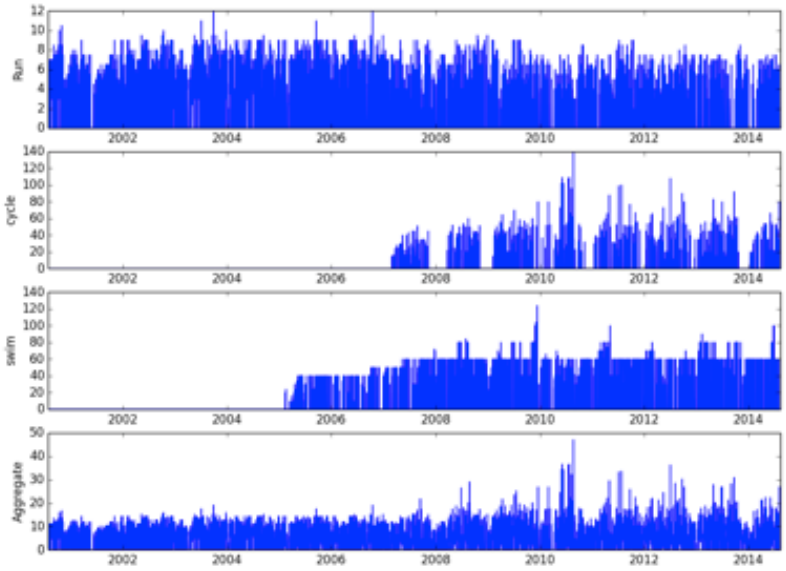


Figure 5-2: Visualization of raw data in the sports activity dataset

In the running, cycling, swimming and aggregated data visualized in Figure 5-2 the X-axis represents time, while the Y-axis is the distance for the corresponding activity. From the visualization, no obvious periodicity can be observed in running, swimming or aggregated data but there seems to be an annual periodicity in the cycling data.

For each sporting activity and for the aggregated data, we applied window sizes of 7, 14, 30, 120, 365 days to calculate the moving averages. Figure 5-3 shows the results of this. Running, cycling and swimming start from 2000, 2007 and 2005 respectively. The moving average calculates the mean value of a fixed size window and then moves the window one day forward to get the new value. Moving average works like a low-pass filter: the bigger the window size, the lower the frequency that can pass. Because of this, it is easier to find long-term trends using a larger window size, since short term shocks in the data (competitions, vacation, short-term injuries) will be smoothed out. From the moving average results, we see that the running distance decreased over time, while the cycling and swimming distances increased. The total amount of energy expenditure according to MET fluctuates, and no obvious trends can be seen in the aggregated data. We can infer from this data that after the athlete started to train for swimming in 2005 and for cycling in 2007, he adapted himself to this by reducing the amount of training for running.

One major difference between the sleep and sports datasets is that the sports dataset has 100% capture rate of activity over 10 years, while the sleep dataset captures just over 80% of the nights in a 2.5 year period. The raw figures on sporting activities are augmented by the athlete annotating most days with text comments that summarise the day and occasionally report on performance or mood. These reports are infrequent (25–30%), and so provide sparse data that we can also examine for periodic patterns.

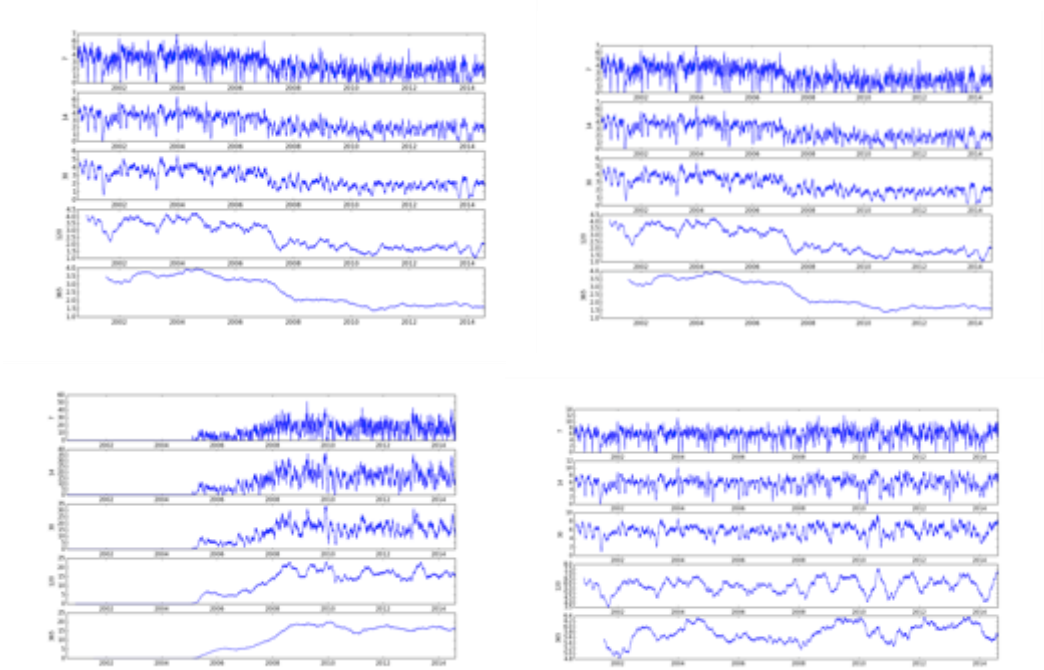


Figure 5-3: Moving average values for sports dataset (Run, Cycle, Swim, Aggregated)

Dem@Care Data

We also examined data captured by the Gear4 sleep sensor in one of the Dem@Care pilot sites. Gear4 is a sleep sensor used to detect users' sleeping conditions. The raw output of Gear4 is a CSV file consisting of 7 data fields, namely recording start time, recording stop time, time fallen asleep, total sleep time, total deep sleep time, number of interruptions, sleep score and full night data (W: awake, S: light sleep, D: deep sleep, A: away). PSD or periodogram cannot be applied directly to full night data, since all the data in the strings are symbols. In order to achieve this, we concatenate calibrated full night data and digitize symbols.

5.3 Results

5.3.1 Periodogram

We applied periodograms and correlations to both datasets to see if periodicities were apparent even with missing data and irregular sampling. The periodogram reveals the energy carried by each frequency across a range and is plotted as a graph where the x-axis is frequency and the y-axis is energy. If there is statistically significant energy carried by one frequency or different frequencies, this will be revealed graphically.

A. Results on Sleep Dataset

Each of the parameters from sleep logging (duration, quality, number of wakes, time in bed, etc.) has been analysed for periodicity but rather than present all of them, we limit ourselves

to just two. For the time asleep, a weekly periodicity is clearly detected as can be seen in Figure 5-4. This can be explained by the weekday/weekend cycle, which is the basis for the subject's lifestyle of working during weekdays and having to get up early to commute to work and then take up leisure activities, waking up later during the weekend. There is also a periodicity around 120-days, i.e. about every 4 months. This cannot be explained without consultation with the subject; an interview session is planned, but had not occurred at the time of this deliverable.

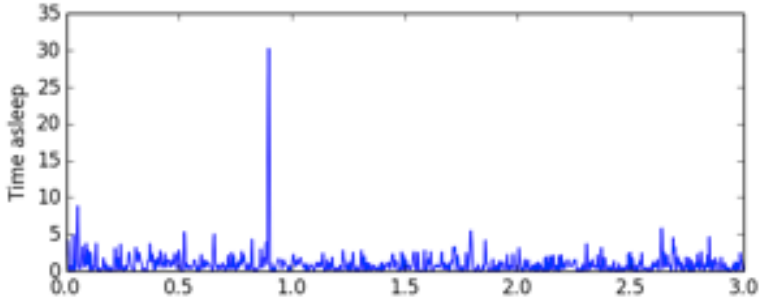


Figure 5-4: Sleep duration periodogram

For sleep quality as shown in Figure 5-5 there is no weekly periodicity. This tells us that even though the subject sleeps more at weekends, he doesn't actually sleep with better quality. We also observe a periodicity around 128 days (ca. 4 months) for sleep quality but at the time of writing, without conferring with the subject, this is something we cannot yet explain.

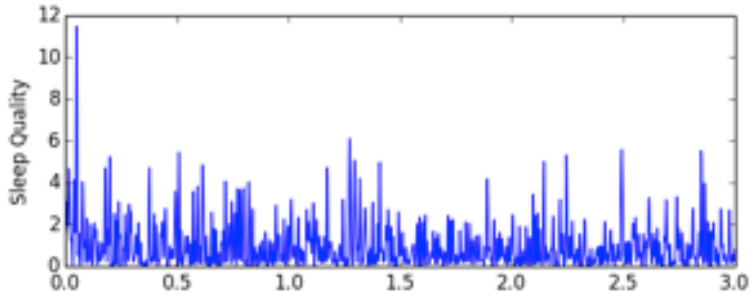


Figure 5-5: Sleep quality periodogram

The other sleep parameters such as time spent in bed, time going to bed have yielded similar results, and so are omitted here. The data here shows that it is possible to detect credible periodicities from lifelogs even though there is missing data and irregular sampling.

B. Results on Sports Dataset

Since the sampling rate of our sports activity dataset is 1 day, the minimum periodic pattern of this dataset we can detect is 2 days. The sports dataset does not have missing data and is consistently and regularly sampled for the three sport activities and for the aggregated data MET levels. In Figure 5-6, periodograms for the sports dataset show interesting results. We can observe three significant energy levels carried by three different frequencies consistently across all 4 subplots. These three frequencies are around 0.14, 0.28, 0.43, corresponding to periods of 7 days, 3.5 days and 2.3 days. Moreover, if we look at the plots more finely, there

exists a frequency at circa 0.0027 located the near the left end of the cycling and aggregated data subplots. This frequency corresponds to the annual period (ca. 365 days) that we observed in the visualization of the cycling data.

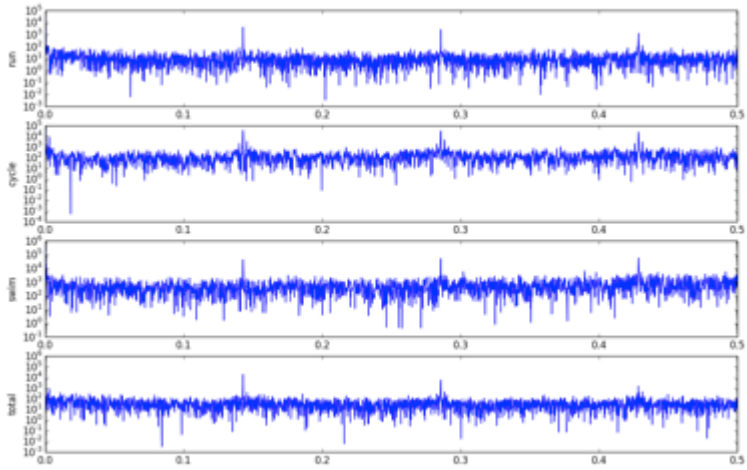


Figure 5-6: Sports dataset periodograms

In order to investigate periodicity in irregularly sampled data, we use autocorrelation. Autocorrelation computes the correlation between the signal and a time-shifted version of the same signal. The x-axis of the autocorrelation plot is time lag and the y-axis is a measure of the correlation of the original signal and lagged signal. If the original signal is periodic then the autocorrelation of the signal should also be periodic and the periods will be located at the peaks of the autocorrelation plot. Autocorrelation of 10 years data is plotted in Figure 5-7. In this graph, there are no periodicities observed in the running, swimming or MET score aggregated data, but an annual periodicity can be found in the autocorrelation of the cycling data. Curious as to where the periodicities over 7, 3.5 and 2.3 days which were found in periodograms from running, swimming and the aggregated data, we took one year of data from 2007 to see if we could detect periodicity in periodograms for just that year. An autocorrelation plot for data from the year 2007 is shown in Figure 5-8.

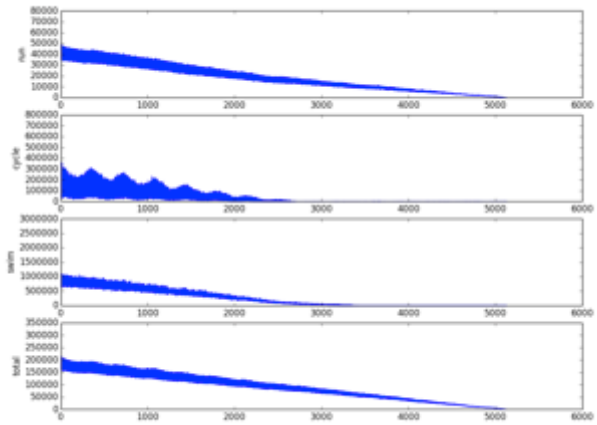


Figure 5-7: Sports Dataset Autocorrelations (10-year span)

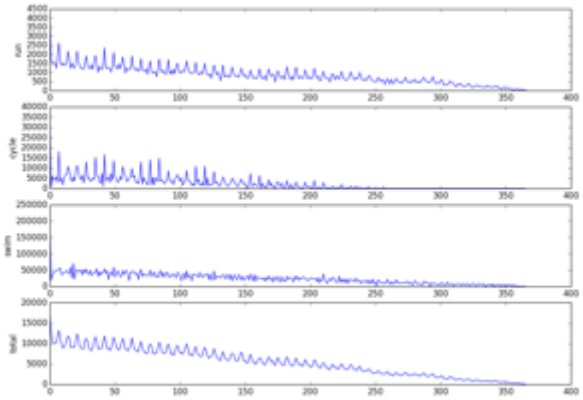


Figure 5-8: Autocorrelation plots of sports data from year 2007

The autocorrelation plot of sports data from 2007 shows that there is a very regular weekly periodicity in running, cycling and in the total energy expenditure of activities, but a less regular weekly periodicity for swimming. We can also find smaller peaks between the two obviously large peaks from running and cycling data, which may correspond to the 3.5- and 2.3-day periodicities also detected in the periodogram. However there are no obvious smaller peaks found in the autocorrelation of aggregated data. A possible explanation may be that these detected periodicities indicate the lifestyle of the subject such as regular scheduled training sessions for running, cycling and swimming. Another explanation might be that there exists an inherent timetable that the subject follows in order to balance participation in the three different activities. For instance the timetable could be every 2 or 3 days run, cycle or swim once. Determining this will require a subject interview, as mentioned previously.

C) Result from Gear4 Sleep Sensor

For the sleep data collected in our @Home pilot site, we calculated periodogram results, shown in Figure 5-9. We can observe in the periodogram that there are several peaks in the result. Spectral leakage is also observed around the peaks. The highest peak is at ca. 36 hours, i.e., about 1.5 days. This result is not typical for sleep data, and requires further investigation. By consulting with the clinicians who collect this data, we will determine if the result shown is as a result of particular aspects of the PwD's sleep patterns, or whether the algorithms here are unstable or otherwise unsuitable for data from this sensor.

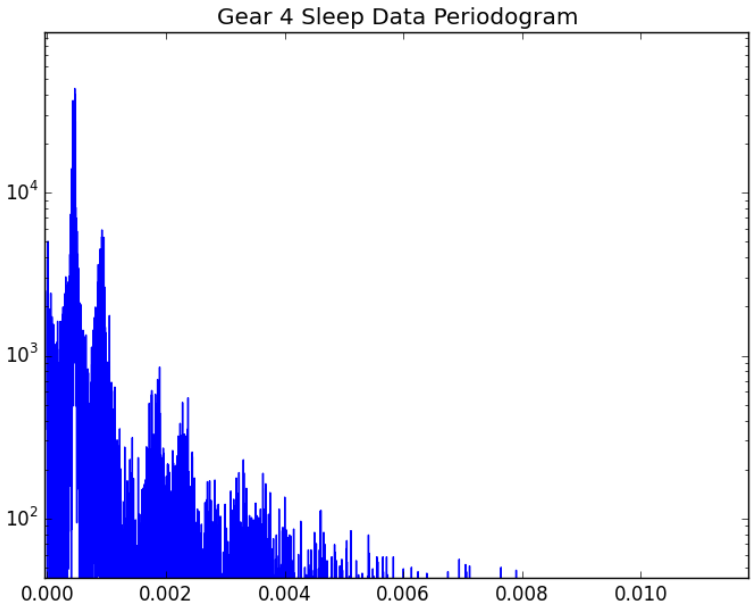


Figure 5-9: Periodogram of @Home pilot sleep data

5.3.2 Intensity of Periodogram

The example showed result of intensity in our running data from athletic database. It is calculated using 35-window size and 28-overlapping size. Figure 5-10 consists of 4 sub-figures: the first sub-graph shows visualize raw running data, Y-axis is the distance the athlete ran, and the X-axis is time scale in years.

The second sub-graph represents total energy carried by each window. It shows a trend of changing total energy. The third sub-graph is plotted by taking energy carried by ca. weekly period and represents intensity of ca. weekly periodicity. We can clearly see a gradually descent in the intensity of the running data and 9th year is a watershed. And from the start of the data to the 9th year, there are peaks and valleys, where we should investigate further to validate whether it is real high/low regularity or false alarms. In the last sub-graph, we can observe a horizontal line at Y-axis equals 7. The sub-graph shows periods that carrying maximum energy against time. The X-axis is time and Y-axis is the periods that have the maximum energy within a window. We can also conclude from the sub-graph that 7-day periodicity is getting less regular. And there are some outliers from weekly line could also be interesting to look into it.

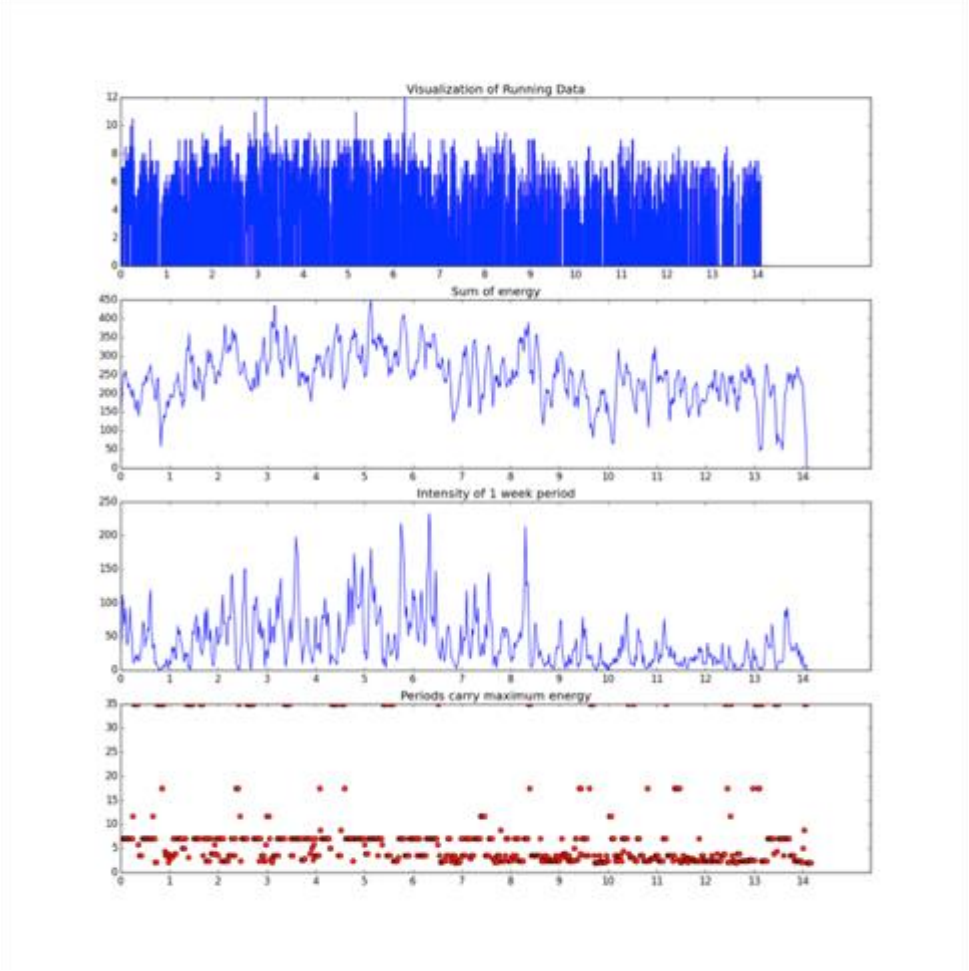


Figure 5-10: Intensity (Running data)

We applied the same algorithms to swimming data and Figure 5-11 shows the result. The sub-graph order is the same as Figure 5-10. One interesting point to be observed from the last sub-graph is that weekly periodicity of running is stronger than swimming.

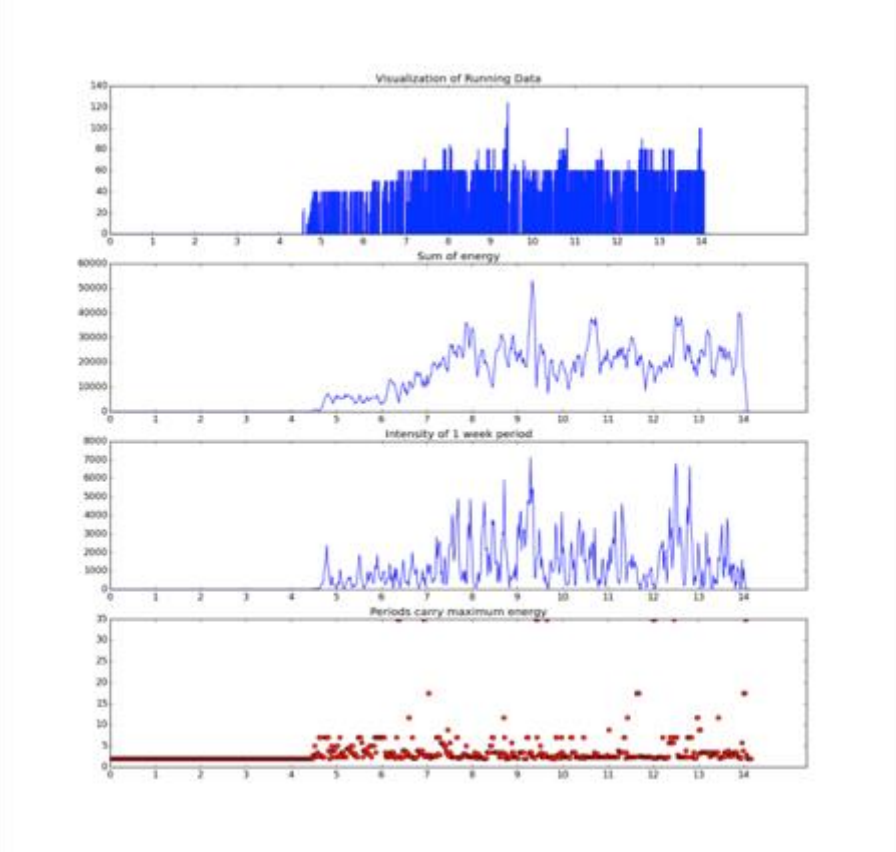


Figure 5-11: Intensity (Swimming data)

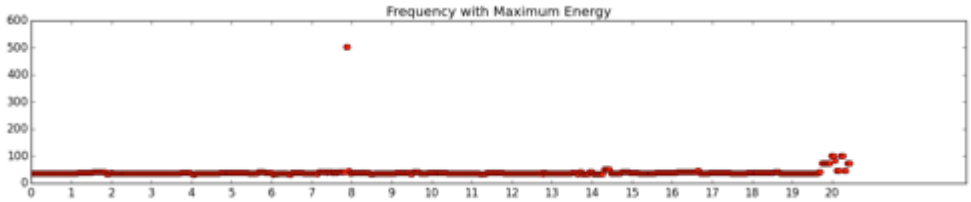


Figure 5-12: Frequency carrying maximum energy (Dem@Care sleep data)

Figure 5-12 shows the frequency carrying maximum energy for sleep data. We can see that the horizontal line resides around 36-hour periodicity. Intensity of 36-hour period is shown in Figure 5-13. Again without getting back to participant, it is hard to understand what happened at the peaks and the valleys.

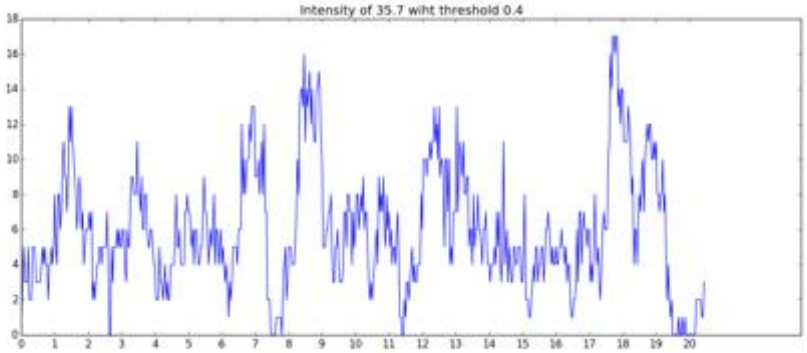


Figure 5-13: Intensity of 36-hour periodicity

5.3.3 Conclusions

In the work presented here, we applied periodicity detection on two longitudinal datasets, which include distances for athletic training and competition for an international triathlete, over a 10-year period, and sleep quality, duration and timing data from a subject over a 2.5-year period. The first dataset was augmented with a pool-based annotation of the triathlete's daily text commentary on his training and performance, from which we were able to get annotations for mood, and for performance. This gave us a collection of datasets which are rich in the variability of their regularity of logging, from consistent and regular daily entries to much more sporadic data with missing data and irregular sampling.

Applying moving average, we discovered that after starting cycling and swimming at a point several years ago, the subject decreased the amount of running while the distances for swimming and cycling kept increasing. The use of periodograms revealed that there are rhythms of repeating patterns at 7, 3.5 and 2.3 days for the running, cycling and swimming data, as well as for when the individual activity data is aggregated based on MET scores. An annual periodicity was also detected in the cycling data. Using an autocorrelation plot for data from year 2007, an obvious weekly periodicity was detected in running, cycling and aggregated MET data but the weekly pattern for swimming is weak suggesting less rigour and regularity associated with training in that sport. An autocorrelation plot of running and cycling shows an unexpected periodicity at a cycle of less than a week (2 or 3 days). This infra-week periodicity may be caused by training schedules for different sports in order to achieve a balanced exercise portfolio. There are no significant periodicities detected in the Lomb-Scargle periodogram for mood or for performance when fused from the annotations of a set of four annotators.

We have demonstrated in this chapter that automatic detection of periodicities from lifelog data can be achieved, even when there is substantial missing data. We have shown that methods based on periodograms and autocorrelation can be used to detect periodicity on complete datasets, while Lomb-Scargle periodograms can be used to detect periodicity on datasets with missing data. Experiments conducted on three datasets with different level of sparsity shows that we are able to detect periodicity in these datasets.

We have also computed intensity of periodicity, and results on different database shows that we are able to indicate regularity of periodicity. Further investigation will be conducted in order to validate the detected peaks, valleys and trends of the intensity of periodicity. We will apply qualitative analysis including interviews with people who provide data (where

appropriate) and/or clinicians who can inform us if the detected intensity and periodicity reflects the actual behavior of the participant who generated the data. This mode of evaluation and verification is necessary when we are dealing with the individual and unique nature of lifelog data.

5.4 References

[5.1] H. Lee, A. F. Smeaton, N. E. O'Connor, G. Jones, M. Blighe, D. Byrne, A. Doherty, and C. Gurrin, "Constructing a sensecam visual diary as a media process," Multimedia Systems, vol. 14, no. 6, pp. 341–349, 2008.

[5.2] A. R. Doherty and A. F. Smeaton, "Automatically segmenting lifelog data into events," in Image Analysis for Multimedia Interactive Services, 2008. WIAMIS'08. Ninth International Workshop on. IEEE, 2008, pp. 20–23.

[5.3] D. Byrne, A. R. Doherty, C. G. Snoek, G. G. Jones, and A. F. Smeaton, "Validating the detection of everyday concepts in visual lifelogs," in Semantic Multimedia. Springer, 2008, pp. 15–30.

[5.4] A. R. Doherty, C. O Conaire, M. Blighe, A. F. Smeaton, and N. E. O'Connor, "Combining image descriptors to effectively retrieve events from visual lifelogs," in Proceedings of the 1st ACM international conference on Multimedia information retrieval. ACM, 2008, pp. 10– 17.

[5.5] P. Wang and A. F. Smeaton, "Semantics-based selection of everyday concepts in visual lifelogging," International Journal of Multimedia Information Retrieval, vol. 1, no. 2, pp. 87–101, 2012.

[5.6] P. Wang and A. F. Smeaton, "Using visual lifelogs to automatically characterize everyday activities," Information Sciences, vol. 230, pp. 147–161, 2013.

[5.7] C. Gurrin, A. F. Smeaton, and A. R. Doherty, "Lifelogging: Personal big data," Foundations and Trends in Information Retrieval, vol. 8, no. 1, pp. 1–125, 2014. http://dx.doi.org/10.1561/1500000033

[5.8] N. Li, M. Crane, and H. J. Ruskin, "Automatically detecting "significant events" on sensecam," International Journal of Wavelets, Multiresolution and Information Processing, vol. 11, no. 06, 2013.

[5.9] M. Vlachos, S. Y. Philip, and V. Castelli, "On periodicity detection and structural periodic similarity." in 2005 SIAM International Conference on Data Mining, vol. 5. SIAM, 2005, pp. 449–460.

[5.10] J. D. Scargle, "Studies in astronomical time series analysis. II - Statistical aspects of spectral analysis of unevenly spaced data," Astrophysical Journal, vol. 263, pp. 835–853, Dec. 1982.

[5.11] M. Jette, K. Sidney, and G. Blumchen, "Metabolic equivalents (METs) in exercise testing, exercise prescription, and evaluation of functional capacity," Clinical cardiology, vol. 13, no. 8, pp. 555–565, 1990.

6 Integration of Components and Usage in Pilots

This section aims to give an overview of the integrated methods that have emerged through this work package, and their use across pilots, interlinking visual sensing research results with integration and clinical piloting (WP7 and WP8). This overview concerns not only the most recent developments presented in this deliverable, but the entire work package, effectively reflecting its total contribution to clinical dementia care.

Table 6.1 captures all methods developed, their inclusion in system modules, integration with the entire system and use in pilots across the consortium. Further details are given below.

The *CAR* (*Complex Activity Recognition*) component is a visual sensing component based on images and depth from ambient 3D-cameras. It integrates the methods described as *Activity Monitoring from RGB-D Camera* (D4.3), *People Detection for Activity Monitoring from Fixed Camera* (D4.4) and *People Tracking for Overlapped Multi-Cameras* (D4.5). Combining the above work it provides: a) location information according to predefined zones in a room, b) activity recognition with the aid of rule-based constructs e.g. according to the duration of one's stay within a zone, zone change patterns, his/her posture, and c) GAIT measurements such as walking speed, distance walked, stride length and number of steps.

Due to its performance and integration, CAR is one of the very few components which provide real-time information in Dem@Care. It has been widely used in @Lab (Nice and Thessaloniki) for activity and gait monitoring and in @NH (Luleå) for sleep, night-time activity and alerts *e.g.* bed exits, restroom visits, falls, among other events.

The *Human Activity Recognition* (*HAR*) module integrates the methods described as *Visual Activity Detection and Recognition* (D4.3, D4.4) and *Activity Monitoring* (D4.5). Despite the advances and improvements in performance, the component's output is still accessible offline. It has been used in @Lab (Nice and Thessaloniki) as an offline result, available after the participant's visit, reaching 77.47% recall and 55.16% precision. It was also used in all the @Home setups in Thessaloniki for kitchen and living room activity recognition, reaching up to 91.33% recall and 3.4% precision.

Multiple clinical results have been derived throughout WP8 deliverables in the various sites for both CAR and HAR, either as stand-alone methods or in combination with semantic fusion.

The *Wearable Camera Processing Unit* (WCPU) integrates the work in *Analysis of Wearable Camera Video*, including object (ORWC) and room recognition (RRWC) (D3.4, D4.4, D4.5), and *Action Recognition* (D5.4, D5.6) (ARWC). Therefore, it is a module capable of offline processing videos from a first-person perspective, and the recognition of locations within an apartment, objects in use and entire composite daily activities. This work has been employed in @Lab in Nice and @HomeDublin, yielding clinical results reported in D8.5. Notably, one of the most interesting aspects of unit consists in the possibility of a precise observation of instrumental activities via close-up views, which would allow for clinicians to identify the

difficulties of patients. This is the usage scenario of the WCPU component explored in the exploitation activities, in the post project exploitation, as detailed in D9.12.

Notably, the visual methods and components in Dem@Care (CAR, HAR and WCPU) may appear to be overlapping, they are in fact used as input to higher-level interpretation. Namely, CAR, HAR and WCPU activity recognition output (also object and location recognition) serve as input to Semantic Interpretation (WP5) which ultimately fuses and merges together atomic events into a definitive activity displayed to clinicians. This is reported in D5.4.

Lifelogging has encapsulated two lines of work: *Visual Analysis in Lifelogging* (D4.3) and *Periodicity Detection* (D4.5). Periodicity Detection and Visual Sensing for lifelogging constitute stand-alone studies, given various restrictions (e.g. lack of APIs) that prohibited their integration to the system within the lifetime of the project. The study proved to be able to detect periodicity by examining longitudinal patterns in sleep, physical activity, and stress data for @Home participants. The analysis also identified the regularity of these patterns (e.g. circadian rhythms were accurately identified for participants' sleep data), and when a different pattern started to emerge. Piloting of this application in @Home in Ireland is detailed in Section 6.4 of D8.5 and its results are presented in the Final Pilot Evaluation Report in sections 6.2.2.1.3, 6.2.2.2.3, and 6.2.2.2.6 of the same deliverable.

Offline Speech Analysis (*OSA*) is the module that encapsulates research in *Voice Analysis for Dementia Assessment and Monitoring* (D4.4). This work has studied clinical interviews either in @Lab, @Home or @NH settings to derive various vocal metrics useful to assessment and care. The works in @Lab have resulted in an integrated OSA component in @Labs, which provides a suggested diagnosis for the individual (reported in D8.4, D8.5). The @Home and @NH studies were not provide an integrated component, but have yielded optimistic results in stress-detection, aiding in long-term observation and care (reported in D8.5).

Table 6.1. Integration of all WP4 components and usage in pilots

Method	**Module**	**Integration**	**Usage in Pilots**				
			@Lab		**@NH**	**@Home**	
			Nice	**Thess**	**Luleå**	**Dublin**	**Thess**
Activity Monitoring from RGB-D Camera (D4.3)	CAR	✔	✔	✔	✔	-	-
People Detection for Activity Monitoring from Fixed Camera (D4.4)							
People Tracking for Overlapped Multi-Cameras							

(D4.5)							
Visual Activity Detection and Recognition (D 4.3, D4.4)	HAR	✔	✔	✔	-	-	✔
Activity Monitoring (D4.5)							
Analysis of Wearable Camera Video (Room and Object Recognition – D4.3, D4.4, D4.5)	WC PU	✔	✔	-	-	✔	-
Action Recognition (D 5.4, 5.6)							
Visual Analysis for Lifelogging (D4.3)	Lifelog	-	-	-	-	(✔)	-
Periodicity Detection (D4.5)							
Voice Analysis for Dementia Assessment and Monitoring (D4.4)	OSA	✔	✔	✔	(✔)	(✔)	-

7 Conclusions

This document presented the research carried out in WP4 (Situational Analysis of Daily Activities) and described the final version of Dem@Care tools aimed at analysing visual data. The performance of the tools was analysed by evaluating their performance on the datasets obtained during data acquisition within Dem@Care in order to perform posture recognition, action recognition, activity monitoring, and life-logging periodicity detection. The improvements in these tools have been presented, compared to their versions presented in D4.2, and compared to the state of the art. Chapter 2 described the research conducted for tracking individuals through a scene using multiple cameras, and showed how the proposed approach improves on the state of the art algorithms for single camera tracking. Chapter 3 presented the research carried out on video analysis for Action Recognition through Object Recognition and Room Recognition on video data from a wearable camera and described the improvements achieved there. Chapter 4 showed the work done for Activity Recognition and Person Detection from video and RGB-D cameras and described how accuracy has improved from earlier versions. Chapter 5 presented Periodicity Detection on longitudinal lifelog data and showed how this can be used to improve the analysis and insight into an individual's regular habits and frequent behaviours.