ECAI 2016
G.A. Kaminka et al. (Eds.)

doi:10.3233/978-1-61499-672-9-1062

Unsupervised Activity Recognition Using Latent Semantic Analysis on a Mobile Robot

Paul Duckworth, Muhannad Alomari, Yiannis Gatsoulis, David C. Hogg, Anthony G. Cohn [1]

Abstract. We show that by using qualitative spatio-temporal abstraction methods, we can learn common human movements and activities from long term observation by a mobile robot. Our novel framework encodes multiple qualitative abstractions of RGBD video from detected activities performed by a human as encoded by a skeleton pose estimator. Analogously to informational retrieval in text corpora, we use Latent Semantic Analysis (LSA) to uncover latent, semantically meaningful, concepts in an unsupervised manner, where the vocabulary is occurrences of qualitative spatio-temporal features extracted from video clips, and the discovered concepts are regarded as activity classes. The limited field of view of a mobile robot represents a particular challenge, owing to the obscured, partial and noisy human detections and skeleton pose-estimates from its environment. We show that the abstraction into a qualitative space helps the robot to generalise and compare multiple noisy and partial observations in a real world dataset and that a vocabulary of latent activity classes (expressed using qualitative features) can be recovered.

1 Introduction

Unsupervised learning over long durations of time has the potential to allow mobile robots to become more helpful, especially when cohabiting human populated environments. Autonomous mobile robot platforms are well suited to continuously update their own knowledge of the world based upon their observations and interactions, using unsupervised learning frameworks. Such robots can be adaptable to their surroundings, the particular time of day, or a specific individual being detected, saving considerable time and effort hard-coding specific information. Understanding what activities occur in which regions and when, allows the robot to adjust its own behaviour, or assist in the task it believes is being undertaken.

The aim of our work is to understand human activities taking place from long term observation of real world scenarios. We present a novel unsupervised, qualitative framework for learning human activities in a real world environment, which is deployed on an autonomous mobile robot platform, seen in Figure 1. The challenge is to learn semantically meaningful human activities by observing multiple people performing everyday activities, and learn a vocabulary which can describe them.

The first main challenge is that each observed activity is likely to be carried out with particular variations, e.g. opening a door with opposite hands. This is called intra-class variation, which our qualitative framework deals with well. A second major challenge is that our robot's on-board sensors only grant our system a partial and mobile view of the world. Using recent advancements in human pose estimation techniques it obtains incomplete and noisy observations of detected humans performing everyday tasks. Our framework helps alleviates these problems by using a *qualitative spatial representation* (QSR) as an effective abstraction method. This allows the system to compare observations based upon key qualitative features and learn common patterns in an abstracted space, instead of their exact metric details which can arbitrarily differ. For example, if a person reaches for a mug on the desk, the exact xyz coordinates of their hand or mug are not important, but the action of approaching and grasping the mug is a useful human activity to learn and understand.

Figure 1. A Metralabs Scitos A5 mobile robot was used to capture and learn human activities in a real-world environment.

To the best of our knowledge, we are the first to combine Latent Semantic Analysis (LSA) [13] with a qualitative spatial representation to recover human activity classes, and a vocabulary to explain them, from a challenging and realistic mobile robot activity dataset. In the following sections, we provide formal details of the qualitative abstractions used throughout the paper, and introduce our activity learning framework methodology. Briefly, the system architecture is shown in Figure 2 and consists of:

1. Detection of humans using an RGBD sensor on a mobile robot; the system estimates and tracks the main skeleton positions. This is introduced formally in § 3.
2. Transformation of the skeleton pose estimates into a qualitative space; qualitative calculi are used to abstract the detected metric coordinates of the person's joint positions. These positions are abstracted first in the camera coordinate frame with respect to the other estimated joints, and secondly, in the map coordinate frame relative to key landmark objects. The qualitative representations used are introduced in § 4.
3. A code book of unique qualitative features is identified by extracting all paths up to some length k, through an interval graph representation of each observation; this is presented in § 4.1.
4. Finally, we use Latent Semantic Analysis (LSA) to recover semantically meaningful (latent) concepts which exist in the feature space. The latent concepts retrieved become our semantically meaningful activity classes which are used to explain human actions. This is presented in detail in § 5.

[1] School of Computing, University of Leeds, UK.
email: p.duckworth, scmara, y.gatsoulis, d.c.hogg, a.g.cohn}@leeds.ac.uk

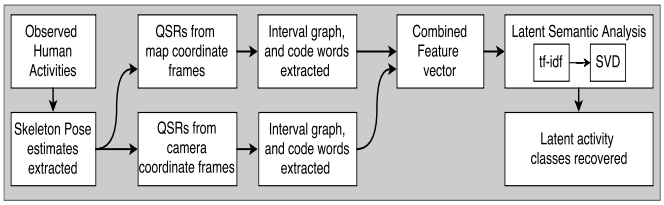

Figure 2. System Architecture.

We build upon a technique used previously [16]; however, in previous work only a single coordinate point of the detected human in the map frame of reference was used to learn common motion behaviours throughout a region of space. We extend this here by adding the estimated skeleton pose of the detected person into the framework, allowing us to generate more detailed and specific activity classes from observations.

Our learning methodology consists of first encoding a video clip (of a detected human) as the occurrences of its qualitative spatio-temporal features which is used as a feature vector to represent the activity. We analyse the collection of feature vectors analogous to a corpus of text documents, looking for semantically similar structures and features (words) that commonly co-occur; these are used to define a vocabulary over which to describe human activities. Instead of documents containing multiple words, our video clips consist of multiple qualitative spatio-temporal features. The activity taking place in the video is akin to learning the document's context. We use *term frequency-inverse document frequency* (tf - idf) [32] scores to weight the observations based upon the importance of the qualitative features observed. For example, if a qualitative feature is present in every video clip, it is given a very low score, as opposed to features which occur less frequently which have a higher score. Finally, we use Latent Semantic Analysis (LSA) [13] to recover latent concepts, effectively semantically clustering qualitative features which commonly co-occur together. This provides information about which qualitative features are used to represent each activity class and is used to define a vocabulary.

We present more information on each of these steps in the following sections, and introduce a new, publicly available human activity dataset captured from a mobile robot in § 6. Analysis and results validating our approach are presented in § 7 and 8, before conclusions are drawn in § 9.

2 Related Work

There is a considerable literature which aims to understand, recognise or detect human motions and activities from video data. There is a long standing field of research in video surveillance, where it is important to be able to track human movements in a particular area. This can be thought of as learning motion patterns in a 2D image plane, and many statistical approaches have been applied [4, 21], along with neural network approaches [22, 24], and also clustering techniques [30]. These approaches make no inference as to what activity the tracked object might be performing, as usually more information is needed about the object to make that conclusion. The research area of activity recognition is broad and its aim is to not only keep track of objects in a scene, but also to draw a conclusion into what they might be doing. More specifically, human activity recognition aims to understand what action a person is performing in the observed scene. There have been many approaches to this task; the majority use data collected from static RGB cameras, but also more recently from RGBD depth sensors. For a more detailed comparison on general activity recognition techniques, the reader is pointed to survey papers which cover the topic using RGB cameras [25, 39, 42] and 3D RGBD cameras [2, 45]. However, similarly to surveillance systems, the majority of approaches in these surveys use a static camera where the field of view does not change and the view point is usually carefully chosen so as to maximise information recorded. The key difference to our work, is that a human activity recognition system deployed on a mobile robot has a changing field of view. This presents a challenging and partial view of the environment, making the observations within a class vary greatly.

Activity recognition from mobile robots is a much more recent field of research, mainly due to the advancements in probabilistic robotics [38]. This has allowed mobile robots to have much more reliable mapping, localisation, and navigation frameworks. Activity recognition using a mobile robot has previously been performed, albeit in a strictly supervised setting. Simple, whole body activities have been learned and recognised using the position and height of a person's detected face [20]. More recently, the locations of estimated skeletal joints have been abstracted using qualitative 3D cone bins to create histograms [44], pose trajectory descriptors [9] and also joint location covariance descriptors [23]. These approaches create a compact and viewpoint invariant representation which is similar to our approach using Ternary Point Configuration Calculus (which is introduced in § 4). A combination of these descriptors are used in [19] on a spontaneous-actions dataset collected from an environment populated by humans, by patrolling a university student area. An SVM learning methodology is used to classify the different activity classes. The key difference between these approaches and our work is that our activity learning is performed in an unsupervised manner. Supervised learning involves annotating each activity observed, and labelling it with the action that is performed before learning can occur. This is unsuitable for a long term patrolling mobile robot since the number of detections obtained is very large. Our method is unsupervised and does not need human input to decide what activity classes to learn. A further advantage is that our methodology is adaptable to changing environments, as it selects the most frequently observed co-occurring qualitative features to define classes.

Similar to a mobile robot field of view, is the recent literature which performs activity recognition from egocentric vision. Qualitative representations have been used in a system to assist with assembling-like tasks from an egocentric perspective [5]; human robot interactions are recognised in [44] using a mixture of skeleton pose estimate features, optical flow and STIP features; early recognition of actions is performed in [34]. However, each of these egocentric approaches use a supervised methodology, which are unsuitable for our long-term autonomous robot.

Many different visual features are used throughout the literature to accurately describe human actions. The authors of [36] define three types of invariance that they use to distinguish between different human actions. They specify that a system should be view invariant, execution rate invariant and finally anthropometric invariant. Our qualitative methodology also satisfies each of these conditions by abstracting and comparing activity observations in a qualitative feature space. The authors present an action classification system which uses action exemplars resulting from projecting the joint angle positions into a subspace, and comparing across frames. Although their technique uses a similar representation of the estimated skeletal positions, it also requires the exemplars of each action in a supervised manner unlike our approach.

There is a large literature which uses qualitative spatial representations to abstract metric visual data. The Qualitative Trajectory Cal-

culus (introduced in § 4), has been used to represent human dancing activities [8]. The authors use a sophisticated infra-red motion-capture system to detect the exact position of each dancer's body which allows them to recover repetitive patterns which are associated with specific dance actions. There is also previous work which uses a distance based qualitative feature, defined by the distance between joints and objects (in particular the floor) [46]. The authors detects abnormal events in daily living activities such as elderly people falling; however this analysis is also performed in a supervised setting, using an SVM and data collected from a static a RGBD camera. A related unsupervised approach uses a qualitative representation to recover repeated events in a static camera dataset consisting of multiple aeroplane turnarounds [37]. The events learned are structured events which occur often between similar tracked vehicles in the dataset, and the view point is static. However, similarly to our work, the authors us qualitative relations to reduce the effect of variations between observations. Qualitative relations are also used with Inductive Logic Programming (ILP) to learn activities from the same aeroplane turnaround dataset [15] but in a supervised manner.

To the best of our knowledge, we are the first to combine LSA with a qualitative spatial representation to learn human activities from a challenging and realistic mobile robot activity dataset. Our qualitative representation abstracts metric observations and takes inspiration from [37, 16]. Previous works have used LSA and pLSA (probabilistic LSA) for learning activity categories in an unsupervised setting, although not from a mobile robot using qualitative features. Approaches have been developed using low-level STIP features [28], local shape context descriptors on silhouette images [47], and a combination of semantic and structural features [43, 26]. These approaches are not performed with the variability of a mobile robot's frame of reference, and were restricted to a single person in the scene during the training phase, unlike ours which can encode feature vectors for multiple people in the scene simultaneously. Further, a major problem cited in the literature is "The lack of spatial information provides little information about the human body, while the lack of longer term temporal information does not permit us to model more complex actions that are not constituted by simple repetitive patterns" [28]. Descriptive spatial-temporal correlogram features have been used previously to attempt to address this issue [35], however, their approach still suffers from low-level image processing frailties, and the requirement for a single person in the scene during training. We address and partially alleviate this problem by using semantically meaningful qualitative features extracted from an interval graph representation. Such features encode more "longer term temporal information" than used in the previous works. Further, our code book of features is adaptable to the environment of the robot, and can contain qualitative relations with semantic landmarks which help understand more complex interactions with key regions, or objects.

3 Skeleton Pose Estimates

Our aim is to understand human activities taking place from long term observations over a varied environment. In this section, we first introduce the input data, followed by the qualitative representation used, and finally describe the auto-generated codebook of qualitative features which results in a term-document matrix representation.

A mobile robot is used to detect humans as they pass within the field of view of its RGBD sensor. The system is hardware independent and modular, although we use an OpenNi skeleton tracker [29] to first detect the human, then estimate and track 15 main positions of the human skeleton, roughly corresponding to 15 joint positions (at approximately 30Hz). An example of this is given in Figure 3 (left), where the estimated skeletal joint positions have been detected using the depth information. Also shown (right) is one region of the global map. The detected person in camera frame coordinates is transformed into the map frame of reference; using the robot's location and orientation of the camera (fitted atop a pan-tilt unit). The global map frame is semantically labelled with key regions and landmark objects in advance, which can be seen as brightly coloured CAD (Blender) models in the image (best viewed in colour).

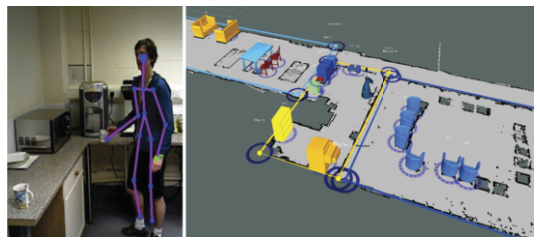

Figure 3. (left:) Skeleton pose estimate at a single timepoint, overlaid onto the original RGB image. (right:) Semantic global map frame.

Formally, we define one skeleton *joint pose* as an xyz Cartesian coordinate in the camera coordinate frame along with a corresponding xyz position in the map coordinate frame, i.e. $j = (id, x, y, z, x_{map}, y_{map}, z_{map})$. A *skeleton pose* then comprises of a collection of joint poses, one for each estimated skeletal joint of the detected human, i.e $p = [j_1, j_2, \dots j_n]$, where $n = 15$ using the OpenNi tracker (i.e. 15 joints are estimated and tracked). For a detected human, we obtain a sequence of skeleton poses over a time series of detections, and generate a *skeleton activity*. This is defined as $S = [p_1, p_2, \dots, p_t, \dots]$, where each p_t is the detected skeleton pose at timepoint t. Note that there are no restrictions placed on t, i.e. each skeleton activity comprises of an arbitrary number of frames and therefore skeleton poses. This depends only upon how long the person is detected by the robot's sensors, and this variation is a major difficulty when using real world data to learn activities on a mobile robot. We discuss how we represent the skeleton data in order to achieve a learning framework next.

4 Qualitative Representation

Abstracting the metric skeleton data using a qualitative representation allows the robot to learn common and repeated activities being performed over multiple observations, even if they vary metrically in their execution. We introduce the qualitative representation used in this paper, before describing the learning methodology in § 5.

Given a video clip containing a human, our system generates a skeleton activity clip, as described above, which comprises of a sequence of skeleton poses (one per frame). Each skeleton pose contains the tracked joints in both a camera coordinate frame and a map coordinate frame. We abstract these exact metric coordinates using a qualitative spatial representation (QSR) expressed in one or more qualitative calculi, which are briefly introduced in this section. The abstraction into a qualitative space allows for the comparison of multiple observations, and the potential to draw similarities which can be used to understand the activities observed. For example, if a person raises their hand above their head and waves, the exact xyz coordinates of their hand or head are not important. It is the relative movement between the two which captures the possible "waving" activity. Likewise, "wave" is an activity which can occur using either hand and as in some earlier literature, the exact joint is abstracted to its joint type when using qualitative calculi; hence the activity is learned

and recognised from observations using either hand (where "hand" is the joint type).

Another reason for representing the data in a qualitative space is that a mobile robot only observes a small section of the world at once from using its on board sensors. The data captured represents only a fraction of the human's total movements in the scene. For example, before a human enters the robot's field of view, the robot has no information about that person, such as which door the person came through, or their intentions. We therefore consider the robot as only partially observing the person's movements. The person also may only appear in the field of view for a few seconds, during which limited time the joint pose estimates can be noisy and inaccurate. Conversely, a person might be performing a static activity and detected for thousands of frames (poses). This variation is a major difficulty, which abstracting the data into a qualitative space helps to alleviate.

In this paper, we use the following qualitative calculi to abstract our activity skeleton data:

1. Ternary Point Configuration Calculus (TPCC) [27]
2. Qualitative Trajectory Calculus (QTC) [14]
3. Qualitative Distance Calculus (QDC) [10]

In these calculi QSRs can be computed from raw xyz data over a series of skeleton poses (frames) using the publicly available ROS library we helped developed [17]. Given recent literature on qualitative representations, all three appear appropriate to describe human actions qualitatively. However, it is not an exhaustive list and other calculi could be explored (something that is out of scope of this work). We briefly introduce and justify each here, with more information available on the QSRLib website [18].

Ternary Point Configuration Calculus (TPCC)

TPCC deals with point-like objects in the 2D-plane. It qualitatively describes the spatial arrangement of an object, relative to two others. e.g. it describes the *referent*'s position relative to the *relatum* and *origin*. This is known as a *relative reference system*, where the origin object is used as an anchor point. In the literature, a 2-dimensional plane is often created from a pair of joints and the relative qualitative location of a third joint is computed. This is equivalent to fixing the origin and relatum to specific detected skeleton joints, and computing the relative positions of each of the other joints. The TPCC reference system is shown in Figure 4. The letters f, b, l, r, s, d, c stand for: front, back, left, right, straight, distant, close, respectively. The implementation details of our use of TPCC are given in § 7.

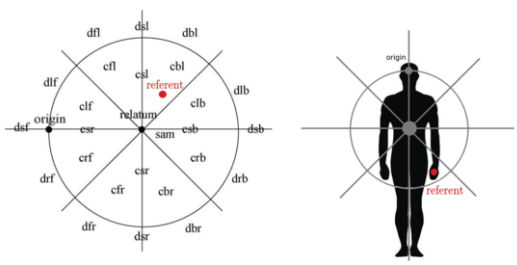

Figure 4. TPCC reference system [27], and overlaid onto a human.

Qualitative Trajectory Calculus (QTC)

QTC represents the relative motion of two points with respect to the reference line connecting them, and is computed over consecutive timepoints. It defines the following three qualitative spatial relations between two objects $o1, o2$: $o1$ is moving towards $o2$ (represented by the symbol $-$), $o1$ is moving away from $o2$ $(+)$, and $o1$ is neither moving towards or away from $o2$ (0). QTC represents relative motion between two objects in a qualitative manner [40] and is considered appropriate for understanding the person's movements in the map coordinate frame relative to key landmark objects. Further details of our use of QTC are given in § 7

Qualitative Distance Calculus (QDC)

QDC expresses the qualitative Euclidean distance between two points depending on defined region boundaries. The threshold boundaries used in this paper are subject to sensitivity analysis, and given in § 7. The intuition behind using QDC is based on the assumption that human motion can be partially explained using distance relative to a key landmark. That is, a set of QDC relations localises a person with respect to a reference landmark, and a change in the QDC relations can help explain relative motion by the person.

4.1 Interval Representation

Many human activities observed by a mobile robot can be explained by a sequence of primitive actions over a duration of time. For this reason, we create a *skeleton activity* from the robot's detections over a number of consecutive poses (of arbitrary length). In this section, we describe how we represent the time series of qualitative detections as a feature vector in a qualitative feature space.

We use Allen's Interval Algebra (IA) [3] to abstract and represent the temporal relations over the observed sequence of QSRs. We abstract the metric skeleton joint coordinates in each skeleton pose using the qualitative spatial calculi above, then compress repeated relations into an interval representation. In the literature, this is equivalent to computing a *Qualitative Spatial Temporal Activity Graph* (QSTAG) from the observed skeleton activity [16, 18]. For example, if the right hand appears to be moving towards the head (QTC relation: $-$), for τ consecutive poses, and then is static (0) with respect to the head for τ' further poses, we compress this into an interval representation consisting of two intervals: $i_1 = \{\text{`}-\text{'}, (0, \tau - 1)\}$ and $i_2 = \{\text{`}0\text{'}, (\tau, \tau + \tau' - 1)\}$, each maintaining the QSR value (or set of values, one per calculi used) and the start and end timepoints. The interval representation of this example is shown in Figure 5 (top row); however an interval representation of a complete skeleton activity contains a single row for each joint (or pairwise joints with objects) that are encoded.

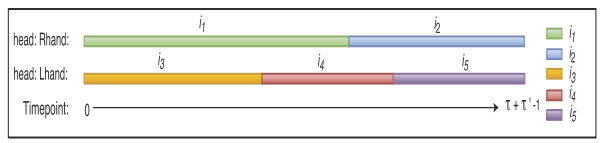

Figure 5. Interval representation of the relations between two skeletal pairwise joints. (Best viewed in colour).

Taking any two intervals in this representation, it is possible to calculate the temporal relation which holds between them using IA temporal abstractions. IA is used to represent and reason with temporal intervals and defines 13 qualitative relations corresponding to seven temporal situations; for two intervals A, B possible temporal relations are: A *before* B, A *after* B, A *meets* B, A *overlaps* B (for a complete list refer to [3]). For example, the IA relation that holds between the i_1 and i_2 intervals is "meets". More detail of how we use this temporal abstraction to create qualitatively meaningful features to describe the video clip is given below.

4.2 Extracting Qualitative Features

Once a skeleton activity video clip is represented as an interval representation, which encodes the sequence of QSRs, we extract a set of unique qualitative features which are used to describe the observation. We define a *code book* as the set of unique features extracted from all observed skeleton activities, defined as $[\gamma_1, \gamma_2, \dots]$, where each γ_i represents one qualitative feature observed in at least one detected skeleton activity. The occurrence of each code word within a skeleton activity allows us to encode a sparse feature vector describing it (with equal length to the code book). This is similar to the *Bag of Words* technique, where words are represented by qualitative features extracted from the videos, and ignores their positional arrangement. This technique facilitates the use of Latent Semantic Analysis (LSA) which is described in § 5.

To extract the qualitative features, we first create the interval representation for each skeleton activity in our dataset as above. We then compute an *Interval Graph* from this representation [12], an example of which can be seen in Figure 6 (encoding both rows present in Figure 5). The timepoints which are implicit in the intervals in Figure 5 are dropped in the interval graph in Figure 6; however the objects and spatial relations involved (e.g. *head, Rhand*,'$-$') for i_1, are still retained in the interval graph (though not explicitly shown in the i'_1 node in Figure 6).

Nodes are linked by directed edges if their intervals are temporally connected, i.e. there exists no temporal break between a pair of intervals. The directed edges are labelled with their IA relation which holds between the two intervals. Thus there is no edge if the IA relation is *before* or *after*. Note, where two intervals occur at the beginning or end of the video clip (and therefore beginning or end of the interval representation), there is insufficient temporal information to abstract over the intervals and there is no edge between these nodes, e.g. there is no edge between i'_1 and i'_3 in Figure 6, as both i_1 and i_3 occur at the start of the observation.

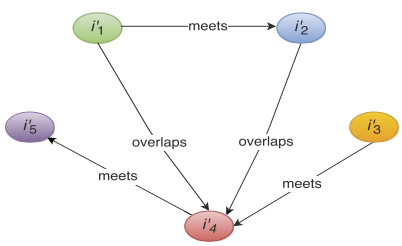

Figure 6. Example Interval Graph.

Code words are generated by enumerating all paths through the interval graph up to and including some fixed length k. This is a new representation for extracting such qualitative code words, although it has been inspired by recent literature [16, 37] [2]. For the interval graph shown in Figure 6, the unique code words extracted (where $k = 2$) are generated by taking all paths up to length 2. This generates the following set of code words: $\gamma_1 = i'_1$, $\gamma_2 = i'_2$, $\gamma_3 = i'_3$, $\gamma_4 = i'_4$, $\gamma_5 = i'_5$, $\gamma_6 = (i'_1 \text{ meets } i'_2)$, $\gamma_7 = (i'_1 \text{ overlaps } i'_4)$, $\gamma_8 = (i'_2 \text{ overlaps } i'_4)$, $\gamma_9 = (i'_3 \text{ meets } i'_4)$, $\gamma_{10} = (i'_4 \text{ meets } i'_5)$. Each observed unique code word, is a path through the interval graph and is equivalent to a *valid graphlet* in the literature [16]. A similar distance based graph kernel is used to encode the code words, for efficient graph comparisons [11].

The code words generated using this technique represent meaningful durations of qualitative relations which were specifically observed within the data. We represent each skeleton activity as a feature vector over this code book, which is therefore an efficient and intuitive method for representing observed human activities. We give the implementation details of the code book that is auto-generated from the mobile robot's observations in § 7.

[2] Interval graphs can be viewed as a representation of a QSTAG [16, 18], in which the temporal nodes are replaced by edges and the object nodes are implicit within the layer 2 spatial episode nodes. We use interval graphs rather than QSTAGs because appropriate code words can be more intuitively expressed.

5 Latent Semantic Analysis

Once each skeleton activity is represented as a feature vector (over the auto-generated code book), we draw comparisons with Information Retrieval systems and use Latent Semantic Analysis (LSA). This is often used to semantically analyse a "term-document matrix", which describes a matrix of word counts over a corpus of documents. In our case, the "terms" are the qualitative spatio-temporal features extracted in our code book, and each "document" in a corpus is a human activity video clip in our dataset. Therefore, creating a feature vector over our code book, for each skeleton activity, generates a term-document matrix.

5.1 Term Frequency - Inverse Document Frequency

Given a term-document matrix D of size $(m \times n)$, (where $m = |dataset|$ and $n = |codebook|$), we apply *term frequency-inverse document frequency* weighting (tf-idf) to scale each qualitative feature by an importance weight depending upon its variation over the whole dataset. It is calculated by the product of two statistics, *term frequency* and *inverse document frequency*. The tf-idf value increases proportionally to the number of times a word appears in a document, and is inversely proportional to the frequency of the word in the entire corpus, which is a measure of how much information that word provides. We use this weighting to adjust for the fact that some qualitative features appear much more frequently in general than others. For example, the QSR value between a person's joints when in a "resting position" will probably appear in the majority of video clips in the dataset. Thus this feature is not informative to learn what activity the person is performing, and will be given a low weighting.

To calculate the tf-idf scores, we use a Boolean term-frequency (tf) weighting and a logged inverse document frequency (idf) weighting:

$$tf(t, d) = 1 \text{ if } t \text{ occurs in } d \text{ and 0 otherwise,}$$

$$idf(t, D) = log \frac{N}{|\{d \in D : t \in d\}|}.$$

where, t is a unique feature (term) in our code book, d is a video clip (document) and N is the number of video clips in the dataset (corpus). Then the tf-idf weights are calculated as:

$$tf\,idf(t, d, D) = tf(t, d).idf(t, D).$$

5.2 Singular Value Decomposition

Once we have the dataset represented as tf-idf weighted feature vectors, the aim is to recover the latent concepts in the data. To do this, we use Singular Value Decomposition (SVD) to extract the singular values, and the left/right singular vectors. This step is key, since the decomposition provides information about the main concepts in the data, along with which features are prominent in each concept.

The technique is akin to finding the eigenvalues of the matrix. A geometric interpretation is that the eigenvalues are the scaling values of the matrix in each specific dimension, whereas the singular vectors are the rotations. It is common to use SVD to obtain a *low rank approximation* of a matrix C, as low eigenvalues are given to vectors which appear as linear combinations of other vectors. This is one of the reasons that LSA helps alleviate the effects of multiple features being synonymous, which can be the case using sparse data. For an $(m \times n)$ tf-idf weighted matrix, the number of non-zero eigenvalues (and therefore singular values), is bounded by the rank of the matrix, i.e. at most $min(m, n)$. However, our aim is to only recover a small number of latent concepts in our matrix, assuming the majority of activities are repeated a number of times. To do this, we make use of the fact that SVD ranks the singular values in non-increasing order along the diagonal matrix Σ.

Given our tf-idf weighted, term-document matrix C, SVD performs the matrix decomposition:

$$C = U\Sigma V^T,$$

where U and V are the singular vectors (rotations around the axis), whilst Σ is a non-increasing diagonal matrix containing the squared eigenvalues of C (i.e. the scaling values in each dimension). Examining the decomposition, the eigenvalues in the diagonal matrix Σ are the latent concepts of the matrix, and can be thought of as the latent activity classes encoded in the data. Further, the columns of the left singular vector (U) contain the eigenvectors of $C^T C$, and hold information about which video clips are assigned to which latent activity class (concept). Finally, the columns of the right singular vector V contain the eigenvectors of CC^T and tell us which qualitative features are used to describe each activity class. This is akin to finding a vocabulary to best describe human activities performed in real world scenarios, in an unsupervised setting.

In the next section we introduce a dataset collected on our mobile robot, and experiments we conducted to evaluate the methodology.

6 Activities Dataset

In this section, we present a new, real-world human activities dataset captured using a patrolling Metralabs Scitos A5 mobile robot, which can be seen in Figure 1. The dataset is captured by observing university members of staff and students performing a set of common every day activities in a real university environment. These are regarded as activity classes. The dataset provides difficult intra-class variation due to different viewpoints and partial occlusions.

The robot is equipped with a laser range finder for mapping and localization and is running ROS Indigo [31] and the full STRANDS system [1]. It is equipped with two RGBD cameras; one chest mounted for the purpose of obstacle avoidance, the other head mounted and used to detect people in the environment using an OpenNi skeleton tracker (introduced in § 3). Given a detected person in the robot's field of view, the camera records RGB images along with the estimated skeleton poses, depth images, plus meta data about the detection i.e. date, time of day, odometry data, region of map [3]. The dataset was collected over the period of one week. The robot patrolled a pre-mapped space which can be seen in Figure 3 (right). The robot's schedule randomly selects between a set of pre-defined *waypoints* to visit and its task once there, is to observe human activities occurring, often for a few hours at a time. During the week, we detected 300 human instances performing 398 daily living activities.

[3] The dataset collected, along with meta-data and software repository, is available at: http://doi.org/10.5518/86.

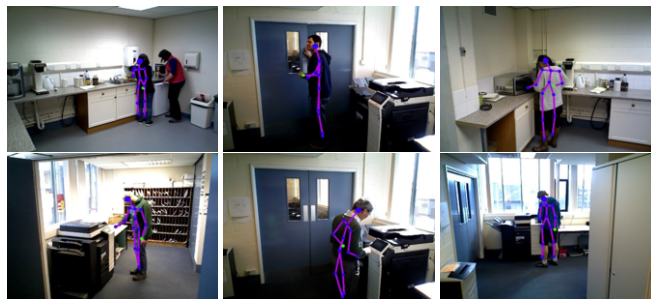

Figure 7. Examples from Human Activities Dataset. The second row are all examples from the same class. (Best viewed in colour).

A selection of example detections can be seen in Figure 7, where the second row, shows three different views of the same activity class (as judged by the ground truth labels).

To reduce any bias, the dataset was annotated by a group of independent volunteers, who segmented each video clip into the human activities occurring. It is these segments which form the basis for the experiments in § 7. No restrictions were applied to the labelling, and the activities could temporally or spatially overlap within the observations. As anticipated, the dataset is unbalanced with respect to the number of each activity class observed (i.e. some activities were observed more frequently than others), and the durations of each instance vary greatly. The following is a complete list of the common activities annotated, along with the number of occurrences: a: Microwave food (17); b: Take object from fridge (52); c: Use the water cooler (26); d: Use the kettle (58); e: Take paper towel (35); f: Throw trash in bin (50); g: Wash cup (66); h: Use printer interface (28); i: Take printout from tray (22); j: Take tea/coffee (35); k: Opening double doors (9). These instances are used during the experiments sections below to highlight the unsupervised learning using interesting human activities.

7 Experimental Procedure

Our experiments comprise one main task; to learn a representation of human activities in an unsupervised setting. For this task, we use the activity instances from the new human activities dataset captured from our mobile robot introduced in the previous section. The main steps of the flow diagram in Figure 2 have already been explained; the implementation details and evaluation metrics are given here, results are presented in § 8, and conclusions drawn in § 9.

Our experiments and results section is presented using the "kitchen" region (defined by the yellow quadrilateral in Figure 3 (right)), which is considered the most interesting in the dataset. In this region there are a total of 10 semantic landmark objects including: *printer, shelves, microwave, water cooler, tea/coffee pot, sink, kettle, fridge, waste bin* and *paper towel dispenser*. For the purpose of evaluation and computational efficiency, we restrict the entire set of 15 skeletal joints to a discriminative subset of 8 including; the *head*, *torso*, left and right, *shoulders*, *hands* and *knees*. For generality when encoding our unique qualitative features below, we do not distinguish between the "left" or "right" *shoulders*, *hands* and *knees*, e.g. the i'_1 node in Figure 6 would contain (*head, hand*,'$-$').

7.1 Qualitative Features

For each activity instance, we generate a skeleton activity and extract QSR features from the metric observations. We do this in a two

stage process, first abstracting the person's relative joint positions in the camera frame, and secondly, abstracting the joint positions relative to pre-defined semantic landmarks. This process generates two sequences of QSR values that are used to create our term-document matrix. The detail of each sequence is discussed below.

7.1.1 Camera Frame

For each skeleton activity S_m in our dataset of M observations, we have a sequence of t skeleton poses. This is referred to as $S_m = [p_1, p_2, \dots p_t, \dots]$, where each p_i contains both camera coordinate and map coordinate frame xyz positions of each joint being used. For each pose in a skeleton activity, we compute TPCC relations for each skeleton joint pose relative to the person's centre line, using the camera coordinate frame, which produces a sequence Q_{cam}. We define a person's centre line by connecting the detected head joint with the torso joint. This equates to fixing the origin and relatum to the head joint pose and the torso joint pose respectively in each frame, and describing the relative position of the referent (i.e. each joint being described). All possible relational values are shown in Figure 4. We obtain a sequence of TPCC relations relative to the centre line Q_{cam}, of length $|S_m|$.

7.1.2 Map Frame

To abstract the person's position in the map frame we use both QDC and QTC relative to key landmark objects in the same region as the detected person. We create a sequence Q_{map} of QSR pairs (QDC and QTC) of length $|S_m| - 1$ (since QTC relies on pairs of consecutive timepoints, we remove the QDC value at $t = 1$ to obtain $|S_m| - 1$ pairs. We also remove $t = 1$ in Q_{cam}.).

QTC was introduced in § 4, but in practice we use the QTC_{B11} variant [14]; since the landmark objects are static in the environment, we capture the relative motion between a joint and an object with just a single QTC_{B11} value, instead of the usual tuple, i.e. the values $(+, 0), (-, 0)$ and $(0, 0)$, are reduced to $(+), (-)$ and (0) respectively. Similarly, the threshold values used for the QDC relations are: *touch* [0-0.25m], *near* (0.25-0.5m], *medium* (0.5-1.0m] and *ignore* (>1m]. Although a comprehensive sensitivity analysis on these values has not been performed, we found the above intuitive considering the locations of the semantic landmarks in some regions are not particularly well spaced out. An example sequence in Q_{map}, *hand-fridge* : [*('+', 'Near'), ('+', 'Near'), ('+', 'Medium'),* ...], of length $|S_m| - 1$.

7.1.3 Generating the Code Book

For each sequence of QSRs, described above, we create an interval representation (by compressing repeated relations) and thus an interval graph. During this process, we apply a median filter which smoothes any rapid flipping between relations, owing to visual noise. By using semantically meaningful QDC relations in the sequence Q_{map}, we do not encode interval graph nodes for any timepoints where the QDC value is "ignore". This has the effect of creating a sparse interval graph, leading to a more efficient process.

To produce our term-document matrix we extract the set of unique qualitative features (code words) by enumerating paths up to some length k, over all interval graphs in our dataset. Since the number of paths increases exponentially with the number of interval nodes, we use $k = 4$ and restrict the nodes on a path to encode at most 4 different objects. During this process, we apply a low-pass filter over the unique paths, stipulating they must be observed a minimum p number of times to be included in our code book; we found $p = 5$ appropriate for our task. This subset allows us to capture overlapping qualitative features, between multiple object pairs which occur in the observations. Finally, since the calculi used in each sequence are distinct, we merge the unique features into a single code book of length 6482 (using the parameters presented above).

Once we have a code book of unique qualitative features, we encode each skeleton activity S_m in our dataset into a feature vector representation, as per § 4.2. We do this by first encoding all the unique code words using a distance based graph kernel for efficient graph comparisons [11]. Secondly, counting the occurrence of each code word in the skeleton activity's interval graph (by comparing to the each extracted path). This process generates a vector of the occurrences of each path in each skeleton activity, which we stack to create the (m x n) term-document matrix D, (where $m = |dataset|$ and $n = |codebook|$).

7.2 Metrics

Since our approach is an unsupervised learning problem and it does not know the labels of each emergent concept, we introduce two popular clustering metrics to evaluate the performance. Our system does not know the label assignments and both metrics provide a score of how closely two sets of labels match (for the same set of data). We use this to compare the ground truth labels (assigned by volunteers), to the emerged concepts from the LSA decomposition. The two metrics are, the V-measure [33] and Mutual Information [41].

The V-Measure is a combination of the *homogeneity* and *completeness* clustering metrics, given two sets of labels. Homogeneity evaluates whether all the predicted clusters contain only data points which are members of the same class; whereas completeness evaluates whether the member data points of a given class are all elements of the same predicted cluster. Both values range from 0 to 1, with higher values desirable. The V-measure is computed using: $v = 2[(\textit{homogeneity} \times \textit{completeness})/(\textit{homogeneity} + \textit{completeness})]$. The second popular metric for unsupervised learning is the Mutual Information score. It can be computed with the following formula:

$$MI(U, V) = \sum_{i=1}^{|U|} \sum_{j=1}^{|V|} P(i, j) log \frac{P(i, j)}{P(i)P'(j)},$$

where $P(i)$ is the probability of a random sample occurring in cluster U_i, and $P'(j)$ is the probability of a random sample occurring in cluster V_j.

8 Results

In this section, we present empirical results that the methodology presented in this paper learns common human activities from unsupervised observations. We demonstrate this by applying our learning methodology, and experimental procedure, to the challenging human activities dataset captured from a mobile robot, introduced in § 6. The structure of this section is as follows; firstly, we evaluate the results of our unsupervised LSA learning framework, using the clustering metrics introduced above. This is supplemented by a comparison to a commonly used supervised learning technique (an SVM) and an unsupervised technique (k-means) used previously in [16]. Secondly, we discuss the learned vocabulary over which each activity class is defined as the occurrences of qualitative features over the code book.

The results presented here are generated by performing LSA onto the term-document matrix D generated in the previous section. This involves applying the tf-idf weights to the term-document matrix D to obtain the weighted matrix C, and performing SVD as per § 5. Figure 8 shows the resulting singular values extracted. It can be seen

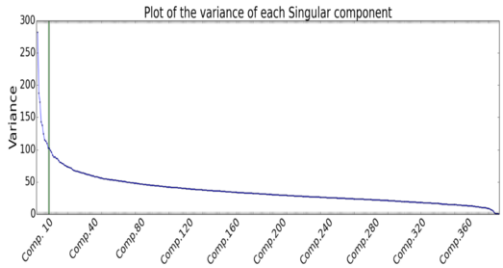

Figure 8. Singular values of LSA decomposition of the weighted, term-document matrix (C). The x-axis represents the singular values, to the maximum of rank(C). Threshold limit for further analysis shown in green.

that there is a limited number of "large" singular values (< 15) where each represents a latent concept in the matrix; this is intuitive given our dataset contains 11 ground truth activity classes. We threshold and use only the largest singular values from Σ (threshold shown as a green vertical line in Figure 8) and the cluster results for 10 latent concepts are presented in Table 1 (we do not present a method for automatically selecting the best threshold in an unsupervised setting in this paper). The dataset contains 11 activity classes; however, the unsupervised system does not know which ground truth labels match each emergent concept.

Table 1 presents a comparison between our unsupervised LSA approach and two popular commonly used alternative methods, a supervised approach (an SVM) which uses the ground truth labels, and an an unsupervised k-means used in previous work [16]. The three methods are compared based upon the same qualitative features. The SVM was trained using 5-fold cross validation, with a linear kernel, and where the code book was trained only once across the whole dataset. It can be seen that the supervised approach obtains 66.1% accuracy on the challenging dataset, and performs only slightly better than the LSA when evaluating using clustering metrics, even though it has access to labelled training instances to create decision boundaries. It can also be seen that LSA outperforms a standard k-means implementation using 10 cluster centres (average result presented over 10 runs, as with random chance classifier). We interpret this as LSA generalising observations better than k-means, since it considers qualitative features with similar meaning, i.e. identifying synonymy between dimensions, unlike k-means.

Metric	LSA	SVM	k-means	chance
V-measure	0.542	0.614	0.368	0.057
Homogeneity Score	0.520	0.617	0.280	0.057
Completeness Score	0.566	0.611	0.542	0.057
Mutual Information	1.180	1.407	0.637	0.130
Normalised MI	0.543	0.614	0.388	0.057
Accuracy	N/A	0.661	N/A	0.113

Table 1. Experimental results comparing LSA, with a supervised linear SVM, unsupervised k-means clustering and random chance clustering.

The results presented demonstrate 10 activity classes are recovered from a challenging, real world, mobile robot dataset. The dataset contains high intra-class variation, shown by multiple view points in Figure 7, and activities that are often occluded and partially observed. The results show the majority of these instances are successfully considered part of the same latent concepts (activity class). This shows the qualitative descriptors used in the abstraction are viewpoint invariant and can handle large amounts of noise and variation during the unsupervised learning phase.

8.1 Learned Vocabulary

As stated in § 5, the LSA decomposition recovers latent concepts in the non-increasing diagonal matrix Σ. In this section, we are interested in the right singular vector V^T. The vectors specify the rotations around the axes whereby the importance of each feature can be determined and a vocabulary defined for each activity class over the auto-generated code book. For our recovered 10 concepts from the above decomposition, V^T (with shape $|codebook|$x10), contains an assignment weight for each qualitative feature (code word), for each latent concept. Two of the right singular vectors from the above decomposition are plotted in Figure 9. We consider these singular vectors as the recovered vocabulary over the latent activity classes present in our dataset.

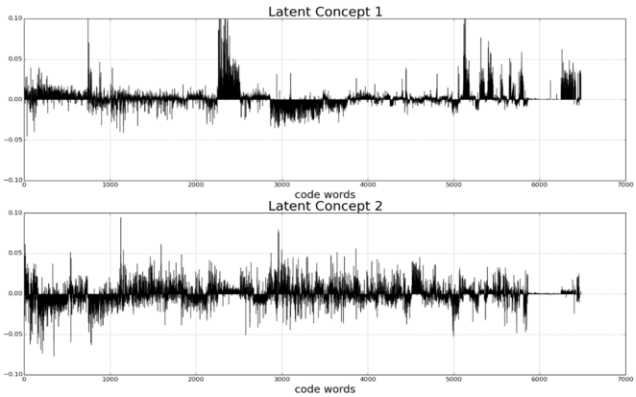

Figure 9. Two latent concepts learned from the LSA decomposition.

9 Conclusion

This paper has presented a novel, unsupervised framework for learning a vocabulary of qualitative features for many common daily living activities, from a mobile robot. We demonstrated its effectiveness at learning ten common activities in a new, real-world human activity dataset with large intra-class variations present.

Our methodology abstracts the exact metric coordinates of a detected person and landmarks, using multiple qualitative representations. It auto-generates a code book from observations, comprising of qualitative descriptors which work particularly well with the occluded and changing field of view afforded by a mobile robot's sensors. Latent Semantic Analysis (LSA) is used to decompose the tf-idf weighted, term-document matrix and recover latent concepts which are regarded as the activity classes observed. The results presented validate our methodology from a challenging dataset and we define a vocabulary for each activity class over the qualitative code words.

Further work includes increasing the complexity and size of the dataset, adding more data pertaining to more varied activities with hierarchical dependencies. This would allow us to learn a hierarchical structure over the activity classes by successively relaxing the number of concepts. We also plan on extending the LSA methodology to a probabilistic framework, using techniques such as pLSA, Latent Dirichlet Allocation [7] or Dynamic Topic Models [6].

ACKNOWLEDGEMENTS

We thank colleagues in the School of Computing Robotics lab and in the STRANDS project consortium (http://strands-project.eu) for their input. We also acknowledge the financial support provided by EU FP7 project 600623 (STRANDS).

REFERENCES

[1] STRANDS project. strands-project.eu, 2016.

[2] J.K. Aggarwal and L. Xia, 'Human activity recognition from 3d data: A review', *Pattern Recognition Letters*, **48**, 70 – 80, (2014).

[3] J. F. Allen, 'Maintaining knowledge about temporal intervals', *Communications of the ACM*, **26**(11), 832–843, (1983).

[4] A. Basharat, A. Gritai, and M. Shah, 'Learning object motion patterns for anomaly detection and improved object detection', in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, (2008).

[5] A. Behera, D. C. Hogg, and A. G. Cohn, 'Egocentric activity monitoring and recovery', in *Asian Conf. on Computer Vision (ACCV)*, (2012).

[6] D. M Blei and J. D. Lafferty, 'Dynamic topic models', in *Proc. 23rd Int. Conf. on Machine Learning*, pp. 113–120. ACM, (2006).

[7] D. M. Blei, A. Y. Ng, and M. I. Jordan, 'Latent dirichlet allocation', *Journal of Machine Learning research*, **3**, 993–1022, (2003).

[8] S. H. Chavoshi, B. De Baets, Y. Qiang, G. De Tré, T. Neutens, and N. Van de Weghe, 'A qualitative approach to the identification, visualisation and interpretation of repetitive motion patterns in groups of moving point objects', *International Arab Journal of Information Technology*, **12**(5), 415–423, (2015).

[9] A. Chrungoo, S.S. Manimaran, and B. Ravindran, 'Activity recognition for natural human robot interaction', in *Social Robotics*, 84–94, Springer, (2014).

[10] E. Clementini, P. Di Felice, and D. Hernández, 'Qualitative representation of positional information', *Artificial Intelligence*, **95**(2), 317 – 356, (1997).

[11] F. Costa and K. De Grave, 'Fast neighborhood subgraph pairwise distance kernel', in *Proc. 26th Int. Conf. on Machine Learning*, pp. 255–262, (2010).

[12] H. N. de Ridder et al. Information System on Graph Classes and their Inclusions (ISGCI). www.graphclasses.org (Interval Graphs), 2016.

[13] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, 'Indexing by latent semantic analysis', *Journal of the American society for Information Science*, **41**(6), 391, (1990).

[14] M. Delafontaine, A. G. Cohn, and N. Van de Weghe, 'Implementing a qualitative calculus to analyse moving point objects', *Expert Systems with Applications*, **38**(5), 5187 – 5196, (2011).

[15] K. S. R. Dubba, A. G. Cohn, and D. C. Hogg, 'Event model learning from complex videos using ILP', in *European Conf. on. Artificial Inteligence (ECAI)*, (2010).

[16] P. Duckworth, Y. Gatsoulis, F. Jovan, N. Hawes, D. C. Hogg, and A. G. Cohn, 'Unsupervised learning of qualitative motion behaviours by a mobile robot', in *Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS)*, (2016).

[17] Y. Gatsoulis, M. Alomari, C. Burbridge, C. Dondrup, P. Duckworth, P. Lightbody, M. Hanheide, N. Hawes, and A. G. Cohn, 'QSRlib: a software library for online acquisition of Qualitative Spatial Relations from Video', in *Workshop on Qualitative Reasoning (QR16), at IJCAI-16*, (2016).

[18] Y. Gatsoulis, P. Duckworth, C. Dondrup, P. Lightbody, and C. Burbridge. QSRlib: A library for qualitative spatial-temporal relations and reasoning. qsrlib.readthedocs.org, Jan 2016.

[19] I. Gori, J. Sinapov, P. Khante, P. Stone, and J. K. Aggarwal, 'Robot-centric activity recognition in the wild', in *Social Robotics*, 224–234, Springer International Publishing, (2015).

[20] D. Govindaraju and M. Veloso, 'Learning and recognizing activities in streams of video', in *Workshop on Learning in Computer Vision, at AAAI*, (2005).

[21] W. Hu, X. Xiao, Z Fu, D. Xie, and T. Tan, 'A system for learning statistical motion patterns', *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **28**, 1450–1464, (2006).

[22] W. Hu, D. Xie, and T. Tan, 'A hierarchical self-organizing approach for learning the patterns of motion trajectories', *IEEE Trans. on Neural Networks*, **15**, 135–144, (2004).

[23] M. E. Hussein, M. Torki, M. A. Gowayyed, and M. El-Saban, 'Human action recognition using a temporal hierarchy of covariance descriptors on 3D joint locations.', in *Int. Joint Conf. on Artificial Inteligence (IJCAI)*, (2013).

[24] N. Johnson and D. C. Hogg, 'Learning the distribution of object trajectories for event recognition', in *Proc. British Machine Vision Conference (BMVC)*, (1995).

[25] G. Lavee, E. Rivlin, and M. Rudzsky, 'Understanding video events: A survey of methods for automatic interpretation of semantic occurrences in video', *IEEE Trans. on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, **39**(5), 489–504, (Sept 2009).

[26] J. Liu, S. Ali, and M. Shah, 'Recognizing human actions using multiple features', in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, (2008).

[27] R. Moratz and M. Ragni, 'Qualitative spatial reasoning about relative point position', *Journal of Visual Languages & Computing*, **19**(1), 75–98, (2008).

[28] J. C. Niebles, H. Wang, and L. Fei-Fei, 'Unsupervised learning of human action categories using spatial-temporal words', *International Journal of Computer Vision*, **79**(3), 299–318, (2008).

[29] OpenNI organization. www.openni.org/documentation, 2016.

[30] C. Piciarelli, G. L. Foresti, and L. Snidaro, 'Trajectory clustering and its applications for video surveillance', in *IEEE Conf. on Advanced Video and Signal Based Surveillance*, (2005).

[31] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, 'Ros: an open-source robot operating system', *Workshop on open source software at ICRA*, **3**(3.2), 5, (2009).

[32] A. Rajaraman and J. D. Ullman, 'Data mining', in *Mining of Massive Datasets*, 1–17, Cambridge University Press, (2011).

[33] A. Rosenberg and J. Hirschberg, 'V-measure: A conditional entropy-based external cluster evaluation measure.', in *EMNLP-CoNLL*, (2007).

[34] M.S. Ryoo, T. J. Fuchs, L. Xia, J. K. Aggarwal, and L. Matthies, 'Robot-centric activity prediction from first-person videos: What will they do to me", in *Proc. 10th Annual Int. Conf. on Human-Robot Interaction*, pp. 295–302. ACM, (2015).

[35] S. Savarese, A. DelPozo, J. C. Niebles, and L. Fei-Fei, 'Spatial-temporal correlatons for unsupervised action classification', in *IEEE Workshop on Motion and video Computing WMVC*, (2008).

[36] Y. Sheikh, M. Sheikh, and M. Shah, 'Exploring the space of a human action', in *10th IEEE Int. Conf. on Computer Vision. (ICCV)*, volume 1, pp. 144–149. IEEE, (2005).

[37] M. Sridhar, A. G. Cohn, and D. C. Hogg, 'Unsupervised learning of event classes from video.', in *Association for the Advancement of Artificial Intelligence (AAAI)*, (2010).

[38] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*, MIT press, 2005.

[39] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea, 'Machine recognition of human activities: A survey', *IEEE Trans. on Circuits and Systems for Video Technology*, **18**(11), 1473–1488, (2008).

[40] N. Van de Weghe, A. G. Cohn, P. De Maeyer, and F. Witlox, 'Representing moving objects in computer-based expert systems: The overtake event example.', *Expert Systems with Applications*, **29**, 977–983, (2005).

[41] N. X. Vinh, J. Epps, and J. Bailey, 'Information theoretic measures for clusterings comparison: Is a correction for chance necessary?', in *Proc. of the 26th Annual Int. Conf. on Machine Learning*, (2009).

[42] D. Weinland, R. Ronfard, and E. Boyer, 'A survey of vision-based methods for action representation, segmentation and recognition', *Computer Vision and Image Understanding*, **115**(2), 224–241, (2011).

[43] S.F. Wong, T. K. Kim, and R. Cipolla, 'Learning motion categories using both semantic and structural information', in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, (2007).

[44] L. Xia, I. Gori, J. K. Aggarwal, and M. S. Ryoo, 'Robot-centric activity recognition from first-person RGB-D videos', in *IEEE Conf. on Applications of Computer Vision (WACV)*, (2015).

[45] M. Ye, Q. Zhang, L. Wang, J. Zhu, R. Yang, and J. Gall, 'A survey on human motion analysis from depth data', in *Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications*, 149–187, Springer, (2013).

[46] C. Zhang and Y. Tian, 'RGB-D camera-based daily living activity recognition', *Journal of Computer Vision and Image Processing*, **2**(4), 12, (2012).

[47] J. Zhang and S. Gong, 'Action categorization by structural probabilistic latent semantic analysis', *Computer Vision and Image Understanding*, **114**(8), 857–864, (2010).