Unsupervised Learning of Qualitative Motion Behaviours by a Mobile Robot

Paul Duckworth
School of Computing,
University of Leeds, UK
p.duckworth@leeds.ac.uk

Yiannis Gatsoulis
School of Computing,
University of Leeds, UK
y.gatsoulis@leeds.ac.uk

Ferdian Jovan
Intelligent Robotics Lab,
University of Birmingham, UK
fxj345@cs.bham.ac.uk

Nick Hawes
Intelligent Robotics Lab,
University of Birmingham, UK
n.a.hawes@cs.bham.ac.uk

David C Hogg
School of Computing,
University of Leeds, UK
d.c.hogg@leeds.ac.uk

Anthony G Cohn
School of Computing,
University of Leeds, UK
a.g.cohn@leeds.ac.uk

ABSTRACT

The success of mobile robots, in daily living environments, depends on their capabilities to understand human movements and interact in a safe manner. This paper presents a novel unsupervised qualitative-relational framework for learning human motion patterns using a single mobile robot platform. It is capable of learning human motion patterns in real-world environments, in order to predict future behaviours.

This previously untackled task is challenging because of the limited field of view provided by a single mobile robot. It is only able to observe one location at any time, resulting in incomplete and partial human detections and trajectories. Central to the success of the presented framework is mapping the detections into an abstract qualitative space, and then characterising motion invariant to exact metric position.

This framework was used by a physical robot autonomously patrolling an office environment during a six week deployment. Experimental results from this deployment demonstrate the effectiveness and applicability of the system.

1. INTRODUCTION

A key factor for the success of intelligent mobile robots, deployed in human populated environments, is their ability to understand human motion. This allows for safer and more effective navigation in populated spaces. Such robot systems perceive and represent the world through a range of sensor modalities, often maintaining abstract representations that allow them to make inferences and decisions. How to best represent knowledge about the world is still a major challenge in the field of intelligent robotics. This problem is magnified on mobile robot platforms where on-board sensors provide only a partial and noisy view of the world.

In this paper, we investigate the problem of how an intelligent mobile robot can learn and predict human motion behaviours in real-world daily living environments (e.g. office areas) from partial and noisy observations of the inhabitants. As such, the problem to be solved is to learn from partial human motions and generalise to underlying motion behaviours.

Appears in: *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016), J. Thangarajah, K. Tuyls, C. Jonker, S. Marsella (eds.), May 9–13, 2016, Singapore.*

Our autonomous mobile robot has a limited field of view due to sensor limitations and occlusions. It detects humans within range of its sensors and uses a robust human tracking algorithm [12] to stitch together detected human positions based on their chronology. We define a *pose* as an xy Cartesian coordinate of a detected human. For each human, our tracking algorithm produces a sequence of poses that we call a *trajectory*. However, these trajectories are always incomplete, representing only a section of the person's motion.

Our assumption is that human motion *relative to key objects* is highly informative of recurrent motion patterns and relates to everyday activities. For example, standing up and walking from a desk towards the printer to collect a printout, is a motion pattern which occurs in many offices, irrespective of the exact xy locations of both the desk and the printer. For all offices, it is difficult to generically express this behaviour in the Cartesian map-plane using a metric approach. However, it is simple to describe this behaviour in relative terms, across offices. Qualitative spatial calculi [6] are well suited to this task as they are able to abstract specific details of observations, extracting similarities whilst preserving qualitative differences. We therefore represent trajectories as a sequence of qualitative spatial-temporal relations (QSTRs) between the human and reference objects. This allows us to abstract away from absolute Cartesian coordinates into a more structured qualitative space. The trajectories we detect have a variable number of metric poses. However, our framework compares their qualitative representations making it highly tolerant to different lengths of trajectories, and suitable for a mobile robot.

The key novelty and contribution of this paper is an unsupervised framework based on qualitative relations that learns and predicts human motion behaviours from a mobile robot's limited view of the world. This approach is able to utilise incomplete and noisy detections, generalising them in a manner that extracts a set of motion behaviour classes.

In the next section we discuss related work, followed by a description of qualitative representations used in this paper. A detailed explanation of our methodology is given in Section 4. We present experiments and results from a six week robot deployment at the UK offices of the G4S security company in Section 5 and 6, followed by our conclusions in Section 7.

2. RELATED WORK

A number of approaches to predict human motion behaviours have been developed particularly in the domain of surveillance. For example, statistical approaches [18, 16, 2], neural networks [17], clustering [22, 21], goal-based state machines [10], etc. The key difference to our framework is that these works use data collected from static cameras with a fixed frame of reference and a wide field of view, which allows them to observe long and complete trajectories. These static camera approaches make no predictions outside their field of view and therefore have limited use outside a surveillance setting, such as in mobile robotics where the field of view is narrow, and often occluded. Motion behaviours over a transportation network have also been learned and predicted from GPS data [20]. Here, similar techniques to the surveillance setting can be used since a person's motion is detected over the entire network. They make no predictions outside of the field of view, and therefore can be considered similar to static camera approaches.

Recent work on trajectory analysis using mobile robots includes [5, 7, 4]. These works have used multiple robots to detect human trajectories. However, their robots are positioned *statically* such that they cover almost the complete region resulting in fully observable trajectories. This requires prior knowledge of where these interesting areas are and pre-defining the robot positioning to obtain the complete trajectories. Furthermore, since they cover the entire region with laser scanners, they segment complete trajectories between 'resting points'. This equates to pre-defining the start and end poses of trajectories, and therefore of potential motion patterns. This is in contrast to our work where a trajectory has arbitrary length and the start can be detected at any point in the global map, which might not be the actual source location of a motion behaviour.

More recently, [19] used a support vector machine to learn trajectory features: speed, area covered, etc. The system is heavily reliant on detecting complete trajectories via a large grid of laser sensors and requires pre-defining feature classes which is not possible in our unsupervised setting.

In this paper we make none of the assumptions found in the papers discussed above. We let our single autonomous mobile robot patrol its operational environment and perform other tasks whilst capturing human detections.

Aside from motion behaviour literature, there is a range of previous work which uses qualitative spatio-temporal relational frameworks similar to those in this paper. These include unsupervised learning of events and event classes using a static camera approach [24], learning activities and classifying online using an egocentric vision approach [3], learning object place arrangements [13], and in a robotic setting using qualitative abstractions in a RoboCup soccer simulator [26]. These works use qualitative frameworks to generalise observations and learn models, each for a different purpose. It is this ability to generalise metric observations into more general models that we draw upon in our work.

3. QUALITATIVE REPRESENTATIONS

Our approach is to characterise movements of people in terms of the qualitative relationships they exhibit with a number of fixed reference points, e.g. doorways, desks, printers, etc. Our framework, based upon qualitative spatial-temporal relations (QSTRs), allows us to keep qualitative information about relative locations, whilst ignoring exact metric location. For example, a qualitative prediction of 'approaching the printer' only needs to state that a person gets closer to the printer over consecutive observations, but does not need to predict precise (and therefore potentially inaccurate) metric changes in position. This notion of 'approaching' provides enough information to allow a robot to reason about its own future actions (e.g. avoiding the region between the human and the printer), and can generalise across printers anywhere in a global coordinate frame. Next we present background work in qualitative representations used throughout this paper, and build upon it in the following sections.

3.1 Qualitative spatio-temporal relations

Given their ability to capture information about trajectories, the following three qualitative relational calculi were considered for our task. All three were computed using a publicly available ROS library we developed [14]:

1. Qualitative Distance Calculus (QDC) [8, 15]
2. Qualitative Trajectory Calculus (QTC) [11]
3. Allen's interval algebra (IA) [1]

QDC: expresses the qualitative Euclidean distance between two points depending on defined region boundaries. In this paper the thresholds are defined as: 'touch' [0-1m], 'near' (1-2m], 'medium' (2-4m], 'far' (4-8m] and 'ignore' (>8m]. The intuition behind using QDC is based on the assumption that human motion can be partially explained using distance relative to key objects. i.e. a set of QDC relations localises a person with respect to reference objects, and changes in these relations can be used to explain relative motion.

QTC: a calculus to represent the relative motion of two points with respect to the reference line connecting them. In this paper, we use the QTC_{B11} variant [11], which relies upon two time points per relation. 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). Since our reference objects are static, we only need to represent the relation between $o1$ and $o2$ rather than the inverse also (as in the full QTC_{B11}). QTC represents relative motion between two objects in a qualitative manner [25] and is considered appropriate for our task. Intuitively people often move towards objects that are of interest and relevant to their motion, whilst objects they directly depart from can also be informative. For example, someone who wants to print a document will most likely follow a motion behaviour *towards* ($-$) a printer.

The two qualitative spatial calculi introduced above complement each other well. QDC describes relative distance between objects, while QTC_{B11} encodes relative motion.

Allen's interval algebra (IA): is a calculus for reasoning with temporal intervals. IA defines 13 qualitative relations corresponding to seven temporal situations; for example for two intervals A, B the possible temporal relations are: A *before* B, A *after* B, A *meets* B, A *overlaps* B, etc. (for a complete list of relations and interpretation refer to [1]).

The benefit of using IA is that it allows us to encode temporal information about states of our chosen qualitative representations abstracted away from an exact instance in time. This allows us to generalise experiences to compare them. We do this by using qualitative spatio-temporal activity graphs explained in the next section.

3.2 Qualitative spatio-temporal activity graphs

For each trajectory, we abstract the sequence of xy poses into a qualitative space using the representations introduced above, and generalise by computing a *Qualitative Spatio-Temporal Activity Graph* (QSTAG) [24]. This abstraction allows whole trajectories to be compared qualitatively. The necessary steps are explained here.

QSR episodes:

Prior to detecting the first trajectory, we manually annotate the global map with semantically meaningful objects, i.e. doorways, desks, printers, etc. An example of a semantic object map, where each object type is a different colour, can be seen in Figure 1 (left). This becomes our set of reference objects O, of known, fixed locations.

Formally, we represent a trajectory, m, as a list of xy Cartesian coordinates, known as *poses*, $P_m = [p_1, p_2, \dots]$. For each trajectory, we generate a set of lists of qualitative spatial relations, where each list encodes the relations between the m poses and a reference object in O, yielding the sequence of QSTRs $Q_{m,O}$. E.g. for a single object in the region, $O = \{desk\}$, we obtain one list of qualitative spatial relations $Q_{m,O} = \{[q_1, q_2, \dots]\}$, where q_i contains one relation from each qualitative calculus used (in our case QDC and QTC$_{B11}$). Adjacent sequences of identical q_i are compressed to form a QSR *episode* (labelled with the start and end time points). For example, if a trajectory relative to a desk yields the following five combined (QDC, QTC$_{B11}$) relations:

$$Q_{m,O} = \{[\{Touch, +\}, \{Touch, +\}, \{Near, +\}, \{Near, 0\}, \{Near, 0\}]\}$$

then we can compress them, maintaining a set of spatial relations and an interval of time over which they hold:

$$Q'_{m,O} = \{[[\{Touch, +\}, (1,2)], [\{Near, +\}, (3,3)], [\{Near, 0\}, (4,5)]]\}.$$

This compression maintains ordinal information about the relations and their durations, without keeping every time point of the observation in memory. The above example can be interpreted as a person initially *touching* the desk, *moving away* until they are *near* the desk, finally *stopping* (with respect to the desk) at a distance of *near*.

We define a QSR episode as a tuple containing the following information: an object pair (a unique trajectory ID and one reference object); the set of qualitative spatial relations, (each belonging to a different qualitative calculus); and an interval of time the relations hold over. For example, the sequence of QDC and QTC$_{B11}$ states above can be represented as three QSR episodes, $E_{m,O} = [e_1, e_2, e_3]$, where:

$$e_1 = (traj\_id, desk, \{Touch, +\}, (1, 2)),$$
$$e_2 = (traj\_id, desk, \{Near, +\}, (3, 3),$$
$$e_3 = (traj\_id, desk, \{Near, 0\}, (4, 5)).$$

Each trajectory observed by the robot is compressed into a list of QSR episodes of variable length, where the length depends upon the number of qualitative spatial relation changes the trajectory experiences with the reference objects. Figure 2 represents the example sequence of qualitative spatial relations as a time-line, where the $x-$axis represents time.

For a region containing j reference objects we obtain $E_{m,O} = [E_{m,o_1}, \dots, E_{m,o_j}]$ representing the set of QSR episodes between the trajectory and each of the objects. Using the entire set of $E_{m,O}$ we generate a corresponding QSTAG g_m, described in the next section.

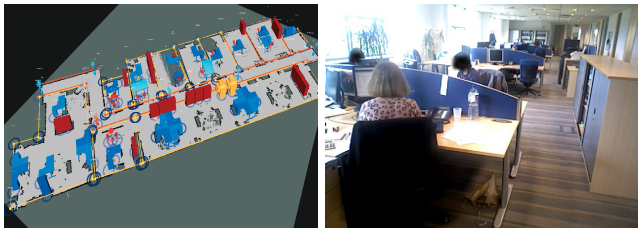

Figure 1: (left) Global map with semantic objects and regions annotated. (right) RGB-image of one region of the deployment map. (Best viewed in colour)

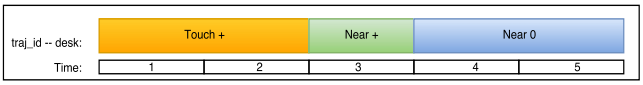

Figure 2: Time-line representation of a trajectory's QSR episode list.

QSTAG:

To generate a Qualitative Spatio-Temporal Activity Graph (QSTAG), from a trajectory, we combine the entire sequence of QSR episodes, $E_{m,O}$. This aggregates the collection of qualitative relations over multiple objects. We do this by generalising the temporal interval of the QSR episodes using Allen's interval algebra. Given $E_{m,O}$ (a list of QSR episodes) as above, an IA relation is computed between every pair of QSR episodes by comparing their intervals, e.g. in the working example, $(1, 2)$ occurs 'before' $(4, 5)$, therefore we say, e_1 occurs 'before' e_3. A QSTAG representing a trajectory ID and a single object can only contain 'meets' and 'before' IA relations, because only one set of spatial relations can hold at any one time between the trajectory and the object. However, a QSTAG representing a trajectory ID and many objects can contain up to seven different IA relations (assuming the arguments to the relations are ordered so as to avoid the six IA inverse relations).

The structure of a QSTAG is composed of nodes partitioned into three layers, and a directed edges set:

- The *objects layer*, contains one node representing the human, and one node per unique object o_j in the list of QSR episodes, where $E_{m,o_j} \in E_{m,O}$. i.e. one node for the human, and a node for each object in O.
- The *spatial episode layer*, contains one node per QSR episode e_i in $E_{m,O}$, where the node encodes the set of spatial relations which hold during that episode.
- The *Allen temporal relations layer*, contains one node per pair of QSR episodes and encodes the IA relation that holds between the two QSR episodes. i.e. a node for each pair, e_i, e_j, where $e_i, e_j \in E_{m_O}$.

The QSTAG for the working example is shown in Figure 3. In general, qualitative spatial calculi contain relations which are asymmetric, therefore we encode the argument order using directed edges (indicated with arrows).

A QSTAG provides a compact and efficient graph structure to represent both qualitative spatial information between moving objects, and the temporal information about those relationships. This facilitates the use of standard graph comparison techniques discussed next.

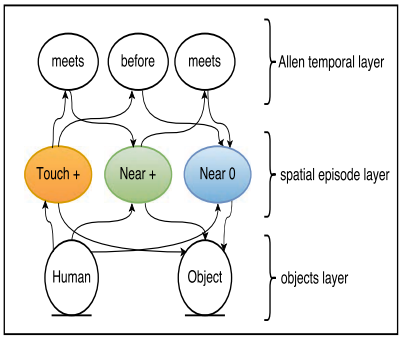

Figure 3: Qualitative Spatio-Temporal Activity Graph between a human and a single object. Directed edges indicate the relations argument order.

Graphlets:

We generate a QSTAG g_m for each detected trajectory m. By comparing multiple QSTAGs we are able to draw conclusions about similarities of the encoded trajectories. A standard technique to compare graphs (such as QSTAGs) is to split each graph into sub-graphs and compare the occurrence of the sub-graphs [9]. However, this requires criteria to be defined on the size and structure of sub-graphs.

We split the QSTAG into a set of overlapping sub-graphs. We restrict the entire set of sub-graphs by defining properties that must hold. This has the effect of restricting the total number of sub-graphs, making the problem tractable, and also only selects sub-graphs that relate to relationships that temporally coincide within a QSTAG. We define a *graphlet* as a connected sub-graph that satisfies the following properties:

i) it maintains the layer structure of a QSTAG;

ii) the objects layer must contain the human node and only one other object node;

iii) the spatial episode layer contains up to three QSR episode nodes which together form a connected interval, this allows the sub-graphs to temporally overlap;

iv) the temporal layer contains up to three IA relation nodes (this is guaranteed by restricting to at most three spatial episode nodes).

This allow us to represent the QSTAG g_m as a bag of overlapping graphlets, $[\gamma_1, \gamma_2, \dots]$, where each graphlet, γ_i, obeys the four properties. The working example QSTAG can be represented as a set of six graphlets, all shown in Figure 4. It can easily be seen that criteria i, ii, and iv hold for each. The third property requires further explanation. A graphlet is restricted to a maximum of three spatial episode nodes to minimise the set of all possible graphlets, whilst allowing them to overlap within the QSTAG. e.g. in the working example, both graphlets γ_2 and γ_3 contain the spatial episode node relating to e_2. Also, a graphlet is said to have a valid duration if it encodes QSR episodes which form a connected temporal interval. e.g. in the example, e_1 and e_2 form a continuous duration over timepoints $(1, 3)$ and hence a valid graphlet, γ_2, is created using these two QSR episodes. The two QSR episodes; e_1 and e_3 do not have a continuous duration, since there is a break at timepoint 3, and therefore there is no graphlet containing spatial episode nodes e_1 and e_3 alone. It is worth noting that γ_1, in the example, is itself g. A QSTAG can be a valid graphlet of itself if it maintains the criteria. The valid duration criteria would become more important if condition (ii) were relaxed to allow more than two object nodes in each graphlet. In that more general case, seven different IA relations would be possible, and the continuous duration property holds for graphlets that encode overlapping QSTRs between multiple object pairs.

The set of all possible unique valid graphlets for a QSTAG depends upon the cardinality of the qualitative spatial calculi being used and the number of objects in the QSTAG.

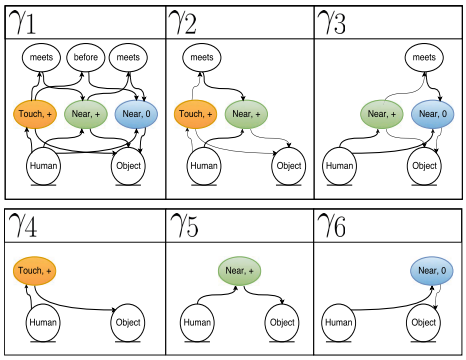

Figure 4: The working example complete bag of graphlets, $[\gamma_1, \dots, \gamma_6]$.

This section presented the theory of QSTAGs and how to extract a set of unique graphlets from a human trajectory. In the next section we explain our methodology for learning human motion behaviours using this representation.

4. METHODOLOGY

The main components of our unsupervised learning framework are shown in Figure 5. It comprises of two phases: training phase only (top layer), and classification phase only (bottom layer). Components common to both phases are shown in the middle layer. More details on these two phases are presented next, followed by a formal description of the framework.

Training Phase:

In the training phase we create a list of QSR episodes for each observed trajectory. i.e. for each trajectory m in our dataset of observed trajectories M, we create E_m, O between the trajectory poses and the reference objects in O. We only encode QSR episodes between the person and the closest n (in this case 5) reference objects (based upon Euclidean distance at the time of detection). This allows us to efficiently capture a spread of relations throughout the region without having to fix the set of reference objects in advance.

For each $m \in M$, a QSTAG g_m is built from the QSR episodes as described in Section 3.2. A set of valid graphlets, $[\gamma_1, \gamma_2, \dots]$ is then extracted from g_m and stored in a database. Therefore, we obtain one QSTAG and corresponding set of graphlets, per observed trajectory.

For all observed trajectories we express the QSTAGs as a set, $G = \{g_1, \dots, g_M\}$, of length $|M|$, where each g_m has a corresponding set of graphlets. We then apply a technique

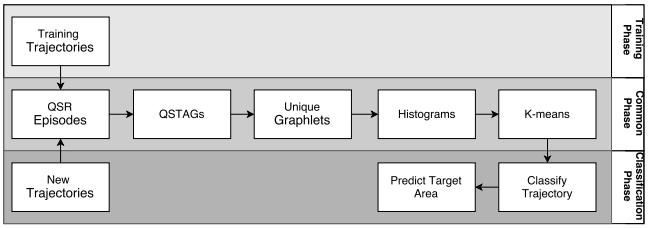

Figure 5: Flowchart of the qualitative unsupervised learning and classification framework (the top layer blocks are for the training phase, the bottom layer blocks are for the classification phase, and the middle layer blocks are common to both phases).

similar to bag-of-words (BoW) to compare the QSTAGs, where each unique graphlet can be thought of as a unique word. This is done by obtaining the set of unique graphlets in all $g_m \in G$, generating a *code book* of words (graphlets) denoted C_G. The code book is incrementally learned and updated over time, depending upon the the training data.

For each QSTAG in G, we count the occurrence of each graphlet code word and create a histogram representation over the code book. i.e. for $g_m \in G$, we compute a histogram h_{g_m} of length $l = |C_G|$. Therefore we represent each QSTAG as a histogram, and generate a set H, where:

$$H = \{h_{g_m} : g_m \in G, m \in M\}.$$

Each histogram in H can be thought of as a feature vector, where each unique graphlet is a unique feature. Using this representation, the exact temporal ordering of each graphlet is lost. However, graphlets that contain two or three spatial episode nodes can overlap within their QSTAG, and therefore maintain ordinal information about the qualitative states within a QSTAG. Further, the representation facilitates the use of unsupervised clustering techniques on H, meaning we can draw comparisons between trajectories in M.

We unit-normalise the histograms in H. We do this because we desire any two histograms with a similar relative proportion of each code word to be considered 'close' in high dimensional space. i.e. if we double every bin of a histogram, after normalisation, the two histograms will be the same. This is a desirable property since different length trajectories can have a different number of QSR episodes and therefore a larger count of graphlets (which tend to be in proportion for the same motion behaviours). However, it is worth noting that a trajectory with fewer poses does not imply fewer QSR episodes. This number depends upon the number of spatial relational changes with the reference objects.

The unsupervised training phase is completed by clustering the dataset of histograms into k clusters, using the k-means algorithm. This produces a model, Θ, with a set of cluster centres, $[\theta_1, \ldots, \theta_k]$, where each θ_i represents a motion behaviour. We determine the value of k, i.e. the number of learned motion behaviours, using the Silhouette Coefficient (SC) [23]. This technique automatically determines k that generates the best model for the data. It uses the mean intra-cluster distance, a, and the mean nearest-cluster distance, b, for each data sample to calculate $SC = (b - a)/\max(a, b)$. The SC value is in the range of $[-1, 1]$, with higher values an indication of better, non-over-fitted, models. As such, with SC we avoid the model over-fitting the data.

Classification Phase:

In the classification phase, the aim is to understand the motion of a newly detected human as quickly as possible. This allows the robot to make a prediction about where that person might be going and alter its behaviour accordingly. The classification process involves comparing a new detection to the learned motion behaviour model, Θ, and as such, is bound by availability of new poses. The people tracker we use groups xy Cartesian coordinates of a person into sub-sequences of poses, to check chronology, where a sub-sequence is a buffer of $0.4s$ of detections and approximately 10 xy coordinates (given a $25Hz$ sampling rate). The classification processes is then repeated for each new sub-sequence of poses.

Once the k-means model is trained and a new trajectory is being observed, the trajectory sub-sequences are available incrementally for the duration of time the human is within sensor range. Given the first sub-sequence, $\approx$10 poses, the common phase steps in Figure 5 are repeated to represent the poses as a QSTAG and as a histogram of unique graphlets. The histogram is unit-normalised and classified into one of k cluster centres, θ_i. This allows the robot to generalise the first sub-sequence, and make an initial behaviour classification using only $0.4s$ of the observed trajectory. It then updates this classification as more trajectory becomes available.

We define the set of data points (training histograms, h_{g_m}) as 'belonging' to their closest cluster centre, θ_i, such that $h_{g_m} \in H_{\theta_i}$, and $H = [H_{\theta_1}, \ldots, H_{\theta_k}]$. Then we can interpret the cluster centre θ_i as the mean histogram of H_{θ_i}, i.e. for $h_{g_m} \in H_{\theta_i}$, θ_i represents the 'influence' of each graphlet γ in the motion behaviour, θ_i. Therefore, when a new trajectory is being observed and is classified into a cluster centre, we use the cluster centre values to extrapolate the trajectory and make a prediction of the QSTRs that we predict to observe in the near future from that motion behaviour.

From the classified cluster centre, the predicted collection of QSTRs are each mapped back onto the metric map plane using the xy coordinates of the reference objects and the QDC relations. This results in aggregating the QSTR predictions over the region, creating a probabilistic *target area* that the robot can use. In practice, this is achieved by maintaining an occupancy grid for each of the k learned cluster centres. The predicted target area is then defined as the most likely occupied group of cells in the grid under a particular motion behaviour. i.e. the target area α_i is predicted from occupancy grid Y_{θ_i}, corresponding to the classified cluster centre θ_i. An example occupancy grid, Y_{θ_i}, is shown in Figure 6.

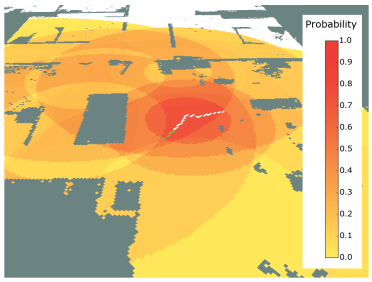

Figure 6: Occupancy grid and predicted target area (yellow-red) from a learned motion behaviour. One trajectory overlaid (green-white) showing sub-sequences of xy coordinates. (Best viewed in colour).

Our training and classification framework presented in this section allows our mobile robot to qualitatively predict human movements within the region of space it is patrolling. The robot can make a decision on which are the most likely qualitative relations a human will achieve with reference objects, continually updating this decision whilst the human is observed. These relations are projected onto the map plane so the robot can either approach the target area to intercept the person, or move away, depending upon the setting.

5. EXPERIMENTS

5.1 Data collection

Data collection took place during a six week deployment at offices of the G4S security company, using a Metralabs Scitos A5 mobile robot. It was equipped with a laser range finder and an RGB-D camera, both of which were used to collect human detections and trajectories. The dataset collected, along with meta-data, is available at: http://doi.org/10.5518/34.

The robot followed a pre-specified schedule during the deployment period which involved patrolling, stationary observation, and object search tasks during weekday working hours (Mon-Fri, 9am-5pm). It was stationary at its charge station during weekends and public holidays. Whilst on its charge station, the robot could perform tasks such as database backups and batch learning, which is when the unsupervised k-means was performed.

During its patrolling and observation operations the robot was detecting and tracking humans. The laser sensor has a 180 degree field of view, however, it was often occluded by obstacles in the environment and fast moving people were only detected briefly. During the six week deployment we recorded approximately $42,000$ trajectories and each was stored as 2D Cartesian poses in a database. Of those, the framework filtered out any with a maximum displacement of less than $1m$, which were considered stationary people or noise. Detections in certain locations were very common (caused by obstacles), i.e. table legs and chairs, being misdetected by the implemented leg detector. After filtering, the average length of a trajectory used in the training phase was 2.2m (median: 1.8m and range: 1m−6.5m). This is in contrast to capturing the complete motion of a person, e.g. a person walking down the approximately 25m long corridor.

The patrolled area was segmented into semantic room regions, each with manually annotated key objects such as desks, bookcases, printers, etc., as shown in Figure 1. In the first week, the robot was predominantly in a particular region that was used for the experimental analysis, resulting in a higher number of trajectories in this region than the following five weeks. This experimental region is the left-most region of the global semantic map in Figure 1 (left).

During the deployment, the robot's behaviour was altered when an observed trajectory was significantly different (based upon a distance threshold), to the learned motion behaviours. The robot would approach the observed person and request them to swipe their security badge.

5.2 Experimental procedure

We evaluate our unsupervised qualitative relational framework using analyses designed to assess the classification and the predicted target area for a new trajectory. Results are presented in Section 6 for the experimental region, which was the most frequently and considered the most interesting region. After filtering, we obtain $1,232$ trajectories in the experimental region with an average length of 2.2m. We use cross-validation of the collected data, where each CV-fold is a different calendar week during the deployment.

One key point is that there is no ground truth with respect to the purpose of a trajectory. Obtaining this would require the robot to either interrupt and ask the human their intention, or an elaborate motion-capture set-up where intention could be inferred from destinations. Neither of these were feasible due to the operational conditions of the deployment. This is a further advantage of the unsupervised framework which learns common motion behaviours which can be post-associated to meaningful activities.

As discussed above, a common problem faced in real world robotic deployments, such as this one, is not being able to perceive complete trajectories from their source locations. This problem is compounded when required to make real-time predictions about a human's future movement, using only the initially observed poses. The people tracker groups the xy poses into sub-sequences as discussed in Section 4. The trajectory poses are stored in the database along with these sub-sequence identifiers. This allows for post-analysis of the trajectories, as if they were being observed live. We use these sub-sequence identifiers during the evaluation to mimic the real-time system.

In our first analysis, we investigate real-time classification metrics to evaluate how well the system classifies a newly observed trajectory into one of the learned motion behaviours θ_i. To do this, for each trajectory, m, in the test set, a QSTAG g_m is generated using only the first sub-sequence poses (≈ 10). The formulation steps are repeated to obtain a histogram representation $h_{g_m}^{[1:10]}$ for this sub-sequence; this is classified into a motion behaviour $\theta_i^{[1:10]}$ and compared against the classification result of the final trajectory θ_m, where all xy poses are used.

In our second analysis, we use the trajectories to evaluate the classification of the k-means model against the prediction of the target area. Here, a previously unseen trajectory, m, is classified into a motion behaviour θ_m as above, and a predicted area is calculated on the metric map using the occupancy grid Y_{θ_m}. An occupancy likelihood score is calculated by taking the average of the occupancy cells corresponding to the xy coordinates of the trajectory. We then check if the likelihood score generated using the classified motion behaviour model θ_m is greater than (or equal to) the likelihood calculated using any other motion behaviour $\theta_i, \forall i \in K \backslash \{m\}$.

Finally, we investigate the effect of adding additional training data into the system. This aims to mimic the live deployed robotic system as it accumulates data over time. Here, we use the final week of data to evaluate, and the first five weeks are incrementally added into the training phase.

6. RESULTS AND DISCUSSION

Figure 7 shows the trajectories that belong to three of k learned motion behaviours in our experimental region. It can be seen that the trajectories belonging to these clusters express specific human behaviours, which we can associate to intent or meaningful activities. For example, the trajectories in the left most image can be interpreted as movements of one specific employee who was present throughout the entire deployment, and whose desk is situated at the source of the trajectories in this image. It shows that this employee

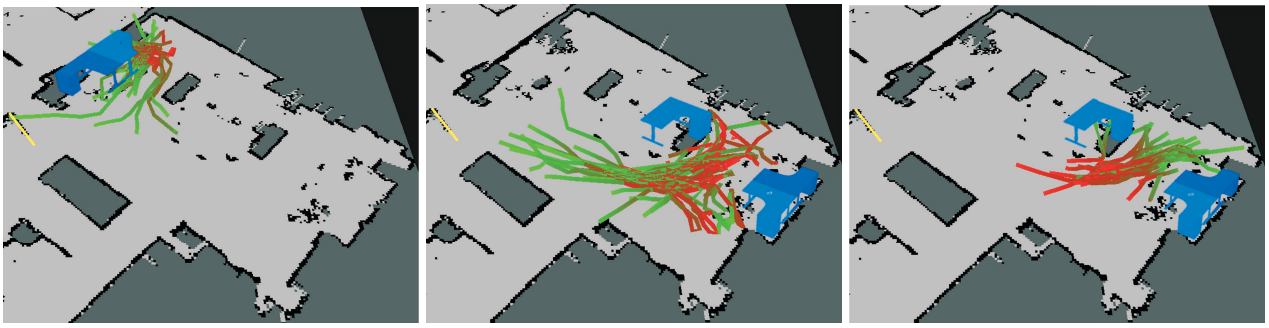

Figure 7: Trajectories belonging to three of the learned k-means clusters in one region, (direction of motion is red to green). Region door showed in yellow. (Best viewed in colour).

commonly walks from behind their desk towards the door (shown in yellow), and we learn this as a common motion behaviour in this region. The centre and right images show there are common motions behaviours towards (right image) and away from (centre image) the collection of desks in the right hand corner of the region. These common behaviours can also be interpreted as employees walking towards and away from their desks. These behaviours enhance our mobile robot's knowledge of the region and each can be translated into a navigation behaviour to approach or avoid the predicted target area when a new trajectory is observed and classified into these motion behaviours.

We evaluate how well the system classifies a newly observed trajectory into one of the learned motion behaviours θ_i. The mean values of recall r, precision p and $F1$-score are calculated after observing increasing percentages of sub-sequences. The results of this first analysis are presented in Figure 8, and were generated using 6-fold cross-validation where each fold is a calendar week of data. The classification process was performed using both QDC and QTC_{B11} relations together, and when using only QDC relations. (QTC_{B11} was not used alone because the QDC relations are needed to predict a target area.) To compare these two classifiers, we collate the data over all percentage-splits and are interested in comparing the number of correctly classified instances (true positives (TP)) with the incorrectly classified instances (false positives (FP)). A 2x2 contingency table is presented in Table 1, where *Test 1* is the classifier based on combined calculi (QDC and QTC_{B11}) and *Test 2* is the classifier based on QDC only relations.

	Test 2 TP	Test 2 FP	Total
Test 1 TP	5498	1640	7138
Test 1 FP	1278	658	1936
Total	6776	2298	9074

Table 1: Contingency table comparing two classifiers based on different QSR calculi.

The graphs in Figure 8 show all metrics are higher for the combination of the two calculi. Further, we perform a McNemar's significance test with null hypothesis that the two classifiers have the same probability of predicting a correctly classified instance. Using a two-tailed test, and a significance level (alpha) of 0.05, we achieve a Z statistic of 6.7, and therefore reject the null hypothesis. This means that the marginal proportions are significantly different from each other and validating our initial belief that QTC_{B11} complements QDC very well. i.e. QDC provides qualitative knowledge about relative distances of the trajectories to objects in the region, whilst QTC_{B11} provides qualitative knowledge about the relative direction of motion.

It can also be seen from the graphs that once 20% of a trajectory is observed, the system has recall $r \approx 0.7$ ($p \approx F1 \approx 0.7$), which demonstrates that even when only a very small section of a trajectory is observed the system is able to perform well. The metrics remain more or less at these values between $20 - 40\%$ of observed trajectory. From 40% and above the metrics increase until all the sub-sequences have been observed.

The results of our second analysis are presented in Table 2. The average scores of recall $r = 0.53$ ($p = 0.67$ and $F1 = 0.55$) demonstrate that, for more than half of new trajectories, the predicted target area generates the highest likelihood score when compared to the other learned motion behaviours. Given the challenging nature of the data (something which is revealed below by the implementation of a previously published algorithm) these results show good performance. This implies that our unsupervised qualitative learning k-means framework is able to express the human motion behaviours that emerge in this environment and predict reasonably well the expectancy of the trajectories using the generated occupancy grids from the cluster centres.

Furthermore, the average recall of the system increases to $r = 70\%$ ($p = 0.80$ and $F1 = 0.72$) if we consider the

cv-fold	k	recall	prec	$F1$
week 0 (342)	13	0.48	0.59	0.48
week 1 (169)	9	0.56	0.80	0.62
week 2 (196)	12	0.48	0.65	0.51
week 3 (104)	10	0.63	0.75	0.67
week 4 (205)	11	0.53	0.53	0.52
week 5 (216)	11	0.48	0.67	0.51
Avg:	11	0.53	0.67	0.55
Std:	–	0.06	0.01	0.08
Random:	–	0.08	0.10	0.08
ZeroR:	–	0.21	0.04	0.07

Table 2: Maximum occupancy likelihood score (testing all k motion models Θ) matching the classified motion.

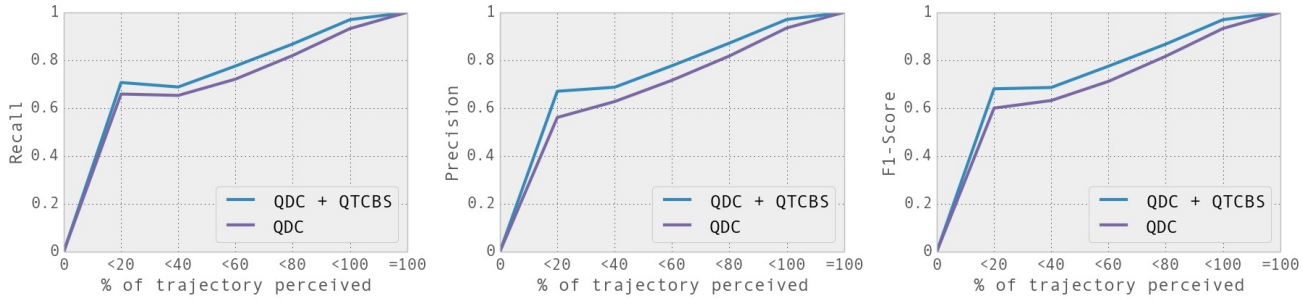

Figure 8: Recall, precision and F1-score presented for the prediction of the final cluster of a new trajectory, given a percentage of its sub-sequences.

highest two occupancy likelihood scores. This is particularly relevant because the predicted target area, given by the occupancy grid Y_{θ_m}, is computed using QDC relations only. However two occupancy grids could appear similar, when their underlying motion behaviour differs due to different QTC_{B11} relations. For example the two motion behaviours, centre and right, in Figure 7 are overlapping with respect to distance relations (QDC) only, but are qualitatively different when considering the direction of motion relations (QTC_{B11}), they are opposite directions.

Finally, we investigate the effect of accumulating training data in a live deployment-like setting. We repeat the previous experiment, but accumulate the training data over five weeks, and test on the sixth. Table 3 shows that as more training data is used, the system is better able to classify new trajectories into motion behaviours. This is further emphasised considering week 0 contained more trajectories than any of the other five weeks.

Training Weeks (M)	k	recall	prec	$F1$
week 0 (342)	9	0.24	0.72	0.29
weeks 0-1 (511)	12	0.43	0.54	0.44
weeks 0-2 (707)	12	0.43	0.56	0.43
weeks 0-3 (811)	10	0.43	0.71	0.49
weeks 0-4 (1016)	14	0.48	0.63	0.53

Table 3: Maximum occupancy likelihood score (testing all k motion models Θ) matching the classified motion, using cumulative training data.

A comparison between our framework and the popular quantitative approach presented in the literature [5], was made by implementing their *Expectation Maximisation* framework to predict motion behaviours. As specified in the abstract of the paper, their objectives appear similar to ours: "This paper proposes an algorithm that learns collections of typical trajectories that characterize a person's motion patterns." It attempts to fit Gaussian Mixture Models (GMMs) over the exact xy locations of trajectories of equal length.

Our training dataset of trajectories were extrapolated to the maximum trajectory length, which in our experiments was 420 xy poses, equivalent to roughly 16 seconds in the robot's field of view. However, the published algorithm was unable to successfully model our data, due to its incomplete and noisy nature. Even using a subset of the training dataset (of 1232 instances) failed to converge in reasonable time, when initialised with a starting number of Gaussian Mixtures. The iterative procedure continued to add a motion model due to low data likelihood, then remove one, due to low motion model utility. This comparison highlights the shortcomings with scalability of quantitative approaches, and the difficulty of clustering real world trajectory data, with no user input.

7. CONCLUSIONS

This paper presented a novel unsupervised learning framework based on qualitative spatio-temporal relations. A qualitative framework that abstracts observations in to a qualitative space is necessary on a mobile robot to generalise incomplete observations. The key challenge was learning motion behaviours and classifying new trajectories given the robot's limited view of the world.

In a high level description, our framework encodes human trajectories into qualitative spatio-temporal activity graphs. Using a BoW-like approach we produce a set of histogram features (which relate to qualitative sub-graphs with certain properties). A k-means algorithm is then trained and queried.

The training and test data were collected over a 6 week deployment of a mobile robot at offices of the G4S security company. Experimental results demonstrate the effectiveness of the system in terms of being able to learn unsupervised meaningful motion behaviours from incomplete trajectories. Analysis using different qualitative spatial calculi showed QDC and QTC_{B11} complement each other well for this task, and provide different modalities of qualitative information about human motions. We also demonstrated that the system is capable of predicting the likely area to be occupied of a newly observed trajectory. Finally, it was shown that the performance increases as it accumulates knowledge.

Future work aims at integrating QTC_{B11} relations into the target area prediction, so a more narrow location can be predicted from the detected human trajectory. Generalising objects to their object types or affordance would allow our system to transfer knowledge between regions. Finally, a challenge is to integrate evaluation metrics to run automatically so that the robot control can benefit from the learned behaviours, forming a closed loop system.

8. ACKNOWLEDGMENTS

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

[2] 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.

[3] A. Behera, D. C. Hogg, and A. G. Cohn. Egocentric activity monitoring and recovery. In *Asian Conference on Computer Vision (ACCV)*, 2012.

[4] M. Bennewitz, W. Burgard, G. Cielniak, and S. Thrun. Learning motion patterns of people for compliant robot motion. *International Journal of Robotics Research*, 24:31–48, 2005.

[5] M. Bennewitz, W. Burgard, and S. Thrun. Using EM to learn motion behaviors of persons with mobile robots. In *IEEE Conf. on Intelligent Robots and Systems (IROS)*, 2002.

[6] J. Chen, A. G. Cohn, D. Liu, S. Wang, J. Ouyang, and Q. Yu. A survey of qualitative spatial representations. *The Knowledge Engineering Review*, 30:106–136, 2015.

[7] G. Cielniak, M. Bennewitz, and W. Burgard. Where is ...? learning and utilizing motion patterns of persons with mobile robots. In *International Joint Conf. on Artificial Intelligence (IJCAI)*, 2003.

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

[9] D. Conte, P. Foggia, C. Sansone, and M. Vento. Thirty years of graph matching in pattern recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 18:265–298, 2004.

[10] H. Dee and D. C. Hogg. Detecting inexplicable behaviour. In *Proc. of British Machine Vision Conference (BMVC2014)*, 2004.

[11] 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.

[12] C. Dondrup, N. Bellotto, F. Jovan, and M. Hanheide. Real-time multisensor people tracking for human-robot spatial interaction. In *Workshop on Machine Learning for Social Robotics at IEEE Conf. on Robotics and Automation (ICRA)*, 2015.

[13] K. S. Dubba, M. R. d. Oliveira, G. H. Lim, H. Kasaei, L. S. Lopes, and A. Tome. Grounding language in perception for scene conceptualization in autonomous robots. In *AAAI Spring Symposium Series*, 2014.

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

[15] Y. Gatsoulis et al. QSRlib: A library for qualitative spatial-temporal relations and reasoning. *In preparation.*

[16] 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.

[17] 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.

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

[19] T. Kanda, D. Glas, M. Shiomi, and N. Hagita. Abstracting people's trajectories for social robots to proactively approach customers. *IEEE Trans. on Robotics*, 25:1382–1396, 2009.

[20] L. Liao, D. J. Patterson, D. Fox, and H. Kautz. Learning and inferring transportation routines. *Artificial Intelligence*, 171:311–331, 2007.

[21] M. Luber, L. Spinello, J. Silva, and K. Arras. Socially-aware robot navigation: A learning approach. In *IEEE Conf. on Intelligent Robots and Systems (IROS)*, 2012.

[22] 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.

[23] P. W. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[24] 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.

[25] 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.

[26] J. Young and N. Hawes. Learning by observation using qualitative spatial relations. In *IEEE Conf. on Autonomous Agents and Multiagent Systems*, 2015.