Intra-Logistics with Integrated Automatic Deployment:
Safe and Scalable Fleets in Shared Spaces

H2020-ICT-2016-2017
Grant agreement no: 732737

DELIVERABLE 2.2

Report on Introspective Active Learning of Site-Specific Knowledge for Intra-Logistics

Due date: month 42 (June 2020)
Deliverable type: R
Lead beneficiary: UoL

Dissemination Level: PUBLIC

Main author: Tom Duckett (UoL), Sergi Molina (UoL)

1 Introduction

This report presents a summary of the work in ILIAD's Work Package 2 for active long-term updating of site-specific knowledge, including scheduling of information-gathering actions to improve map quality and knowledge of activity patterns, together with quantification of the quality of the predictions made and evaluation of performance during long-term operation in dynamic warehouse environments (Task 2.4). The research has been jointly developed by partners University of Lincoln (UoL) and Örebro University (ORU), with UoL being the main beneficiary.

The main objective of WP2 is to ensure long-term operation of the ILIAD system. The system should maintain and update its representations of the environment over time and learn site-specific information for each particular logistics warehouse. While the spatio-temporal models presented in Deliverable 2.1 like the STeF-map [1] or CLiFF-map [2] enable the robot to learn statistics of environmental changes such as flows of people, exploration strategies to efficiently and intelligently build and maintain these models are also required.

For example, the STeF-map results of Molina et al. [1] were obtained by assuming full observability in space and time. In other words, all people detections that provide the input for constructing the STeF-map model of motion patters were captured by sensors that covered the entire operational area under consideration. However, realistically speaking, the situation where a robot is able to observe the whole environment at all times without any occlusions is very rare. Usually a mobile robot is only able to observe a certain region of the operational area at a given time, due to the limited field of view of its sensors and occlusions by other objects. Since the robot cannot be in two places at once, this means that while the robots is gathering information at one location, activities happening at other locations will remain unseen. Models like Kucner et al. [2] handle the spatial data sparsity by applying imputation methods such as Monte Carlo or Nadaraya-Watson. The results show that Monte Carlo tends to preserve the multimodality of the data, while Nadaraya-Watson smooths the data and introduces gradual changes between data points. However, while imputation methods can help to fill in missing data values, they will not address the general problem of exploring unknown parts of the environment.

Furthermore, a robot cannot usually employ 100% of its time for exploration activities. This time is limited by the fact that a robot usually has to complete other tasks, as per its overriding mission requirements (not least for professional service robots in production or intralogistics), or has to take time out to recharge its batteries. So the robot can only devote a certain amount of time during the day for data gathering purposes.

Therefore, in Section 2, a new methodology for mobile robot exploration is introduced to maximise the knowledge of human activity patterns, by deciding *where and when* to collect observations [3]. Also, a new metric based on the chi-squared distance is introduced to evaluate the performance of the exploration (Section 2.3.4).

2 Spatio-temporal exploration for human motion patterns

This section addresses the problem of creating a spatio-temporal pedestrian flow model as accurately as possible from sparse observations of non-trivial environments, taking into account the limited sensory capabilities and time constraints of a mobile robot. This is done through the introduction of an exploration policy driven by the entropy levels of the cells in a STeF-map computed from previous observations [3].

As reported in the previous deliverable D2.1, the Spatio-Temporal Flow map (STeF-map) is a time-dependent probabilistic map able to model and predict the flow patterns of people in the environment. The representation models the likelihood of motion directions

on a grid-based map by a set of harmonic function, which efficiently capture long-term variations of crowd movements over time. The experimental evaluation presented in D2.1 showed that the model enables a better human motion prediction than spatial-only approaches and an increased capacity for safe and socially compliant robot navigation.

The main contribution reported here is the investigation of a new methodology for mobile robot exploration to maximise the knowledge of human activity patterns, by deciding where and when to collect observations based on an exploration policy driven by the entropy levels in a STeF-map. The evaluation is performed by simulating mobile robot exploration using real sensory data from three long-term pedestrian datasets, and the results show that for certain scenarios, the proposed exploration system can learn STeF-maps more quickly and better predict the flow patterns than uninformed strategies.

The work presented extends the methodology first presented in Molina et al. [4]. There, a fixed 24-hour period for human activities was assumed, which is not always the case. In the work presented here, this assumption is relaxed and instead spectral analysis is performed in the time domain for each modelled cell in the environment model. Moreover, in the experiments carried out, a clustering algorithm is used to segment the environment into regions, the number of periodicities for each cell are learnt independently, and the fidelity of the simulation is increased by modelling robot navigation in the environment.

2.1 Problem definition

Assuming an indoor environment with a known spatial layout, which is divided into a set of S square cells forming a grid, the main aim of the robot is to learn a spatio-temporal model of each cell that best represents the human motion patterns, i.e. minimising the error between the internal model and the true human motion distributions over time.

In order to simulate a robot's limited visibility, let us define a region R_i as a subset of s_i number of cells that can be observed simultaneously, such that by visiting R_i during an interval of time $[t, t+\Delta t)$, the observations of human motion within only the s_i cells that belong to R_i are obtained. The remainder of the cells $S - s_i$ in that interval remain unseen. The observations are performed over a predefined interval of time because, as opposed to other environmental variables such as the state of an occupancy cell (free or occupied), the distribution of human motion can not be obtained in a single instant of time, since the robot needs to count a sufficient number of detections to build a meaningful distribution. In the experiments carried out in this section, the regions are assumed to be non-overlapping such that $S = \sum_{i=1}^{\rho} s_i$, where ρ is the total number of regions R into which the environment has been divided.

Although some improvement in model accuracy can be achieved by visiting the regions as often as possible, the number of observations is typically limited and the robot can spend only a fraction of the total time on actual exploration. In the experiments this fraction is referred to as the exploration ratio e. For example, $e = 0.25$ means that the robot can spend only 25% of its operational time on gathering information.

Therefore, given an exploration ratio e and a set of τ non-overlapping time intervals $[t_j, t_j + \Delta t)$, it is the exploration strategy's job to define which regions to observe and in which time interval, complying with the e ratio defined, in order to improve the accuracy of the spatio-temporal human motion model as much as possible.

2.2 Defining when and where to explore using entropy

In this subsection, the definition of information entropy adopted in this work is explained, together with the meaning of entropy in a cell, and a description of the entropy-based policy for deciding where and when to explore.

2.2.1 Entropy definition

It is reasonable to think that cells in diverse parts of the environment could present different motion patterns at different times, so the entropy is used as a measure to define how predictable or unpredictable those patterns are.

Taking, for example, a single randomly chosen cell c with N total people occurrences, its probability distribution for the human motion orientation is defined as $P(X) = \{x_1 = n_1/N, x_2 = n_2/N, ..., x_k = n_k/N\}$, where the count vector $n = \{n_1, n_2, ..., n_k\}$ accumulates the observed occurrences of each orientation bin k. Following the definition in the previous section, the entropy associated with a given cell is:

$$H_{\text{cell}}[P(X)] = -\sum_{i=1}^{k} P\big(\frac{n_i}{N}\big) \cdot \log_2\big(P\big(\frac{n_i}{N}\big)\big). \tag{1}$$

Equation 1 yields the correct answer as N/k tends to infinity, but in many practical cases, this estimator is significantly biased, since $P(X)$ is calculated from a finite set of data. To mitigate this issue, the first-order Miller and Madow correction ([5]) is applied,

$$H_{\text{cell}}[P(X)] = -\sum_{i=1}^{k} P\big(\frac{n_i}{N}\big) \cdot \log_2\big(P\big(\frac{n_i}{N}\big)\big) + \frac{k-1}{2N} \cdot \log_2(e). \tag{2}$$

Although this estimate still retains some bias when $N << k$ or $N \sim k$ [6], this is not the case for this application as the number of people detections tends to be greater than the number of bins. This correction adds more entropy to the cells with fewer detections, expressing the fact that having fewer data to define the motion models indicates that the distribution obtained can be trusted less.

In this context, the Shannon entropy of a cell defines how randomly people move in different directions. Lower entropy values indicate that people tend to follow well-defined motions, while the higher the entropy becomes, the more randomly people move across the boundaries of that cell.

2.2.2 Defining when to explore

This subsection explains how, given a set of observable τ intervals occurring in the future, the entropy-based temporal policy decides which ones to explore while complying with the exploration ratio e. The underpinning idea is to use the map entropy from previous data gatherings to compute the entropy for the future τ time intervals, defining the chances of each interval being chosen for exploration purposes. The steps proposed to do so are given as follows:

I) After each interval of time in which a data gathering action has been performed, the map entropy is calculated as the sum of all cell entropies:

$$H_{\text{map}}([t, t+\Delta t]) = \sum_{c=1}^{S} H_{\text{cell}_c}[P(X|[t, t+\Delta t))], \tag{3}$$

where $H_{\text{cell}}[P(X|[t, t+\Delta t))]$ is given by Equation 2 but the human motion counts are obtained during a specific interval of time. For example, Figure 1 shows the entropy values $H_{\text{cell}}[P(X|[t, t+\Delta t))]$ calculated using non-overlapping time intervals of 1 hour for a randomly picked cell.

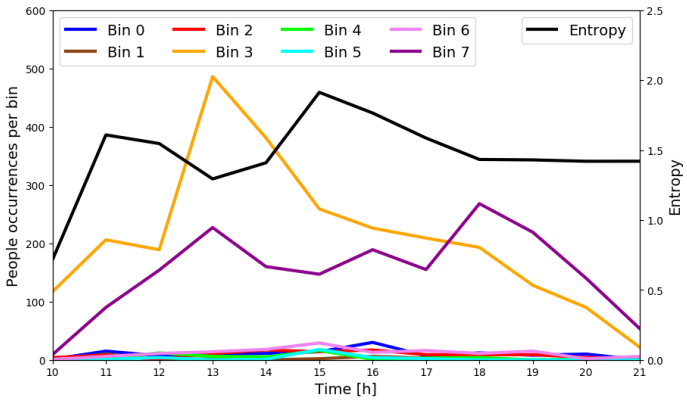


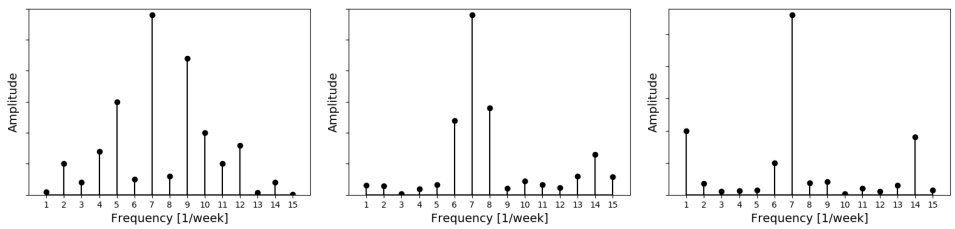
Figure 1: Entropy calculated over 1 day using the distributions obtained in a cell in 1 hour time intervals.

II) Treating the H_{map} as a signal over time, the next step is to compute the most prominent time correlation in the entropy values. To do so, a spectral analysis is performed. Since the map entropy input values are not equally sampled in time due to the time constraints (when $e < 1$), the Non-Uniform Discrete Fourier Transform (NUDFT) is used for this analysis. Notice that the number of input values used to calculate the NUDFT are determined by the exploration ratio defined. The higher the value of e, the more inputs are available to compute the spectra. This frequency decomposition tells us which is the most prominent time correlation over the data by checking the periodicity T with the biggest amplitude (discarding frequency 0). This can be then used to compute the averaged entropy of an interval following the correlation obtained as:

$$H_{\text{int}_j}([t_j, t_j + \Delta t)) = \frac{\sum_{z=1}^{(t_j - t_{\text{start}})/T} H_{\text{map}}([t_j - z * T, t_j + \Delta t - z * T))}{(t_j - t_{\text{start}})/T}, \quad j \in 1, 2, ..., \tau, \qquad (4)$$

where t_{start} is the time when the exploration activities started.

For example, Figure 2 shows the spectra of the three environments used in the experiments after gathering data for some time. Taking the example of the office dataset, the most prominent time dependency is 24 hours (frequency = 7 [1/week]). Knowing this value, for instance, the entropy measures for a 24-hour period corresponding to a day divided into 10 minutes intervals are computed by averaging the entropy values of the same 10-minute interval across multiple days (e.g. from 10:10 to 10:20 from day 1, 2, 3, ...). Note also the peak with a frequency of 1 week (frequency = 1 [1/week]), which appears due to the different people behaviour between weekdays and weekends.



(a) ATC (b) Corridor (c) Office

Figure 2: Frequency spectra of the three environments map entropy values.

III) The probabilities that define the chances of each time interval to be chosen for data gathering are calculated as

$$P(\text{int}_j) = 1 - \frac{H_{\text{int}_j}([t_j, t_j + \Delta t))}{\sum_{z=1}^{\tau} H_{\text{int}_z}([t_z, t_z + \Delta t))}, \quad j \in 1, 2, ..., \tau, \tag{5}$$

so that an array $I = P(\text{int}_j), \quad j \in 1, 2, ..., \tau$ containing the probabilities of all the intervals can be obtained. Finally, using I, the temporal entropy-based scheduling is determined by Algorithm 1.

input : I, e, τ
output: Q (array of 1s and 0s defining the intervals to explore)

begin
  Initialise Q to all 0s
  NumberOfIntervalsToExplore $\leftarrow \tau \cdot e$
  while sum(Q) $\neq$ *NumberOfIntervalsToExplore* **do**
    $\text{int}_{\text{chosen}} \leftarrow$ Choose an interval(I) `// the chances of an interval being picked are proportional to its probability`
    if $Q[\text{int}_{\text{chosen}}] == 0$ **then**
      $Q[\text{int}_{\text{chosen}}] = 1$
    end
  end
end

Algorithm 1: Choose intervals to explore

2.2.3 Defining where to explore

This section explains how, given a set of ρ regions R, each containing a subset of s observable cells, the entropy is used to calculate the probabilities of each region R_i being picked (in the intervals chosen by Algorithm 1) by the entropy-based spatial policy for data gathering purposes. As opposed to the approach presented for deciding when to explore, in this case, the temporal entropy evolution is not taken into account. Instead the entropy is calculated from all the accumulated people detections gathered in the past. In order to calculate the probabilities for each R_i the steps proposed are as follows:

I) First, the entropy of each region R_i is computed as the sum of the entropy of all the cells that are observable (s_i) from the R_i as

$$H_{R_i} = \sum_{c=1}^{s_i} H_{\text{cell}_c}[P(X)], \quad i \in 1, 2, ..., \rho, \tag{6}$$

where $H_{\text{cell}}[P(X)]$ is given by Equation 2 from all the detections seen in the past distributed over the k bins.

For example, Figures 3 and 4 show the occurrences for 2 different cells (a and b) using $k = 8$ discrete orientations in 1 hour time intervals and their corresponding cumulative distributions. The cell a in Figure 3 shows a distribution with two clear peaks corresponding to the two domination orientations of human motion, corresponding to a low entropy value ($H_{\text{cell}_a} = 1.58$). By contrast, Figure 4 illustrates a cell b in another part of the environment where each one of the $k = 8$ orientations obtains a similar number of people detections, corresponding to a more unpredictable behaviour, obtaining a flatter distribution, and hence a higher entropy value ($H_{\text{cell}_b} = 2.76$).

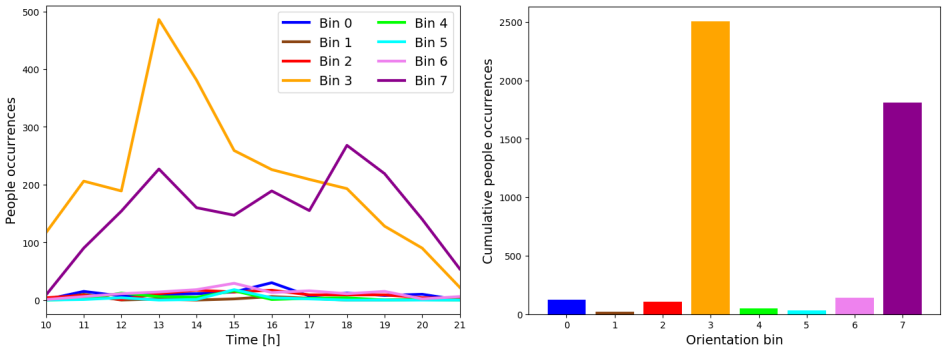


Figure 3: Pedestrian counts in each direction over a day and the cumulative distribution for the cell a with low entropy.

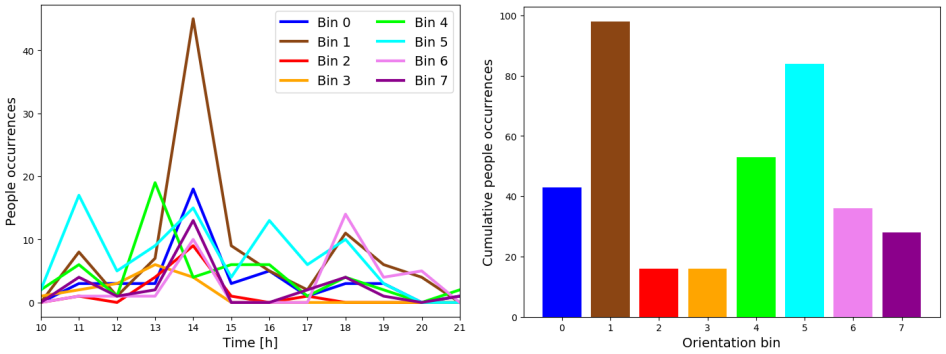


Figure 4: Pedestrian counts in each direction over a day and the cumulative distribution for the cell b with high entropy.

II) Second, the probabilities that define the chances of each region R_i to be chosen to explore are calculated as

$$P(R_i) = 1 - \frac{H(R_i)}{\sum_{j=1}^{\rho} H(R_j)}, \quad i \in 1, 2, ..., \rho. \tag{7}$$

So that every time a certain interval is chosen to be explored, the region where the robot will travel to obtain information about the human motion has to be defined. Similar to the approach presented for the intervals in Algorithm 1, this decision is chosen randomly, but the probability of selecting a given region R_i is proportional to $P(R_i)$.

For example, let us take an environment with just two regions with each region containing just one cell. R_1 has only the cell a from Figure 3 and R_2 has only the cell b from Figure 4. After gathering the information shown in Figures 3 and 4, the associated probabilities in this case would be $P(R_1) = 1-(1.58/(1.58+2.76)) = 0.64$ and $P(R_2) = 1-(2.76/(1.58+2.76)) = 0.36$.

2.3 Evaluation

2.3.1 Experimental scenarios

In the experimental section, three exploration policies are tested (Entropy, Random and Round Robin), which define the set of rules to create the exploration sequence both in time and space.

– **Entropy (E)** policy: the regions/intervals are chosen following the schemes presented in the previous section. The recalculation of the entropy levels is done at the end of each day of exploration using all the data gathered on that day.

For the two uninformed cases implemented, the environment dynamics are not taken into account. These strategies calculate the sequence of visits simply from the number of intervals τ, the number of regions ρ and the ratio e.

– **Random (R)** policy: as its name indicates, the regions and intervals for exploration are chosen in a uniformly random way. Namely, all the τ intervals and all the ρ regions have the same probability.

– **Round Robin (B)** policy: all the areas/time intervals of the environment are visited with the same frequency, interleaving the observations so that the exploration ratio e is satisfied.

In initial comparisons, a **Greedy (G)** algorithm was also implemented [7]. This always looks for the intervals/regions with the highest probability defined by Equations 5 and 7, respectively. However, in all the experiments this method performed poorly compared to the other three policies aforementioned. For that reason, the results for this approach are only presented in Figure 13 but not in the rest of the figures containing the box-plots. This might seem a bit counter-intuitive, but the fact that the robot always goes to the area with the lowest entropy is not always the best option, as there is no chance for the other parts to be explored. This is due to the fact that in this experimental scenario, observing a cell does not imply a change in its entropy after the measurement, as the entropy is only a measure that defines the type of patterns that can be found in a given cell or time interval.

The Entropy, Random and Round Robin exploration policies are compared using the 9 different spatio-temporal combinations: R-R, B-R, E-R, R-B, B-B, E-B, R-E, B-E and E-E, where the first letter indicates the policy for choosing the location and the second letter indicates the one in charge of deciding when to explore. So, for example, in the case of R-B, the region to explore is chosen in a random way, while the time to explore is deterministic.

Regarding the temporal aspect, three different exploration ratios are used: $e = 0.5, 0.25,$ 0.125. This percentage defines the number of time intervals that the robot will devote to gathering data from the total number of intervals available.

2.3.2 Datasets

To evaluate the approach, we ran the experiments using three real pedestrian datasets. All feature complex human movement and enough days to train the models and evaluate the different exploration strategies in the long term. The pedestrian detections in the environments are given in x, y coordinates together with the angle of movement α for every timestamp t. From each dataset we have taken certain days for training, some for validation, and others for testing, but none of the days for each set overlap.

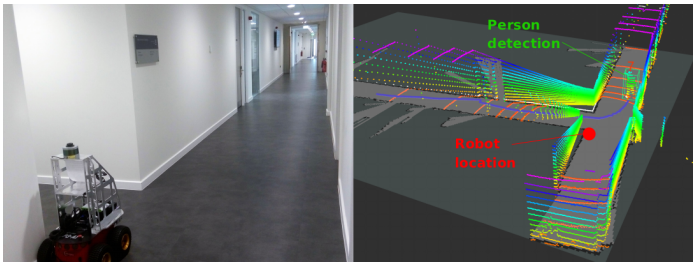


Figure 5: Corridor dataset: Robot location in the corridor and example of a person walking seen by the Velodyne scans.

Shopping centre: ATC

The first dataset was recorded by tracking pedestrians at the the **ATC** shopping centre in Osaka, Japan [8]. The perception system consists of multiple 3D range sensors covering an area of approximately 900 m^2, which is able to detect and track all the people at every instant of time. The data was recorded on every Wednesday and Sunday between October 24th, 2012 and November 29th, 2013, resulting in a total of 92 days. From these data, we selected the first 46 consecutive days (23 Wednesdays and 23 Sundays), using 42 to perform exploration, 2 days for validation and the other 2 for testing. The recording of each day provides people trajectories starting from approximately 09:00 until 21:00, so for the rest of day we assume there are no occurrences of people, simulating the shopping centre being empty. Each of the recorded days contains around 1 million detections of people.

Corridor

The second dataset was collected at one of corridors in the Isaac Newton Building at the University of Lincoln. The data was recorded by a Pioneer 3-AT mobile robot equipped with a 3D lidar (Velodyne VLP-16) and a 2D lidar (Hokuyo UTM-30LX). During data collection, the robot remained stationary in a T-shaped junction, which allowed its sensors to scan the three connecting corridors simultaneously, covering a total area of around 75 m^2 (see Fig. 5). However, since the robot could not stay at the corridor overnight due to safety rules, and it was needed by other researchers occasionally, we did not collect the data on a full 24/7 basis. Instead, the data collection was performed in 10-12 hour sessions starting before the usual working hours. Recharging of the batteries was performed overnight, where the building is vacant, and there are no people on the corridors. The resulting dataset is composed of 14 data-gathering sessions recorded over a span of four weeks. From these, 10 days were used for training, 2 for evaluation, and the remaining 2 for testing. To detect and localise people in the 3D point cloud provided by the lidar, we used an efficient and reliable person detection method developed by Yan et al. [9]. A typical session contains approximately 30,000 detections of people walking in the monitored corridors.

Office

The third dataset was also collected in the Isaac Newton Building building at the University of Lincoln, but in this case, inside a large open-plan office (Figure 6). The recordings were done with a static 3d lidar (Velodyne VLP-16) placed on a tripod at 1.8 meters height and using the same people detector as in the corridor dataset [9]. The sensor was placed in a position covering two entrances to the office, an open area and the coffee area, covering an

Figure 6: Photo of the office area covered by the Velodyne VLP-16 recordings.

area of approximately 85 m^2. The dataset contains 22 days recorded consecutively starting November 23rd, 2018, and each day contains around 25,000 entries during working hours, which are usually from 08:00 to 20:00. In the experimental section, these days are divided using the first 18 for training, the following 2 for validation, and the final 2 for testing.

2.3.3 Model parameters

In the experiments, the space is discretised into 1×1 m cells for all environments, resulting in a total of $S = 1248$, $S = 117$ and $S = 126$ active cells, respectively. The number of bins chosen to discretise the orientations in all three cases is $k = 8$, distributed as shown in Figure 7.

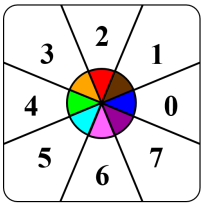
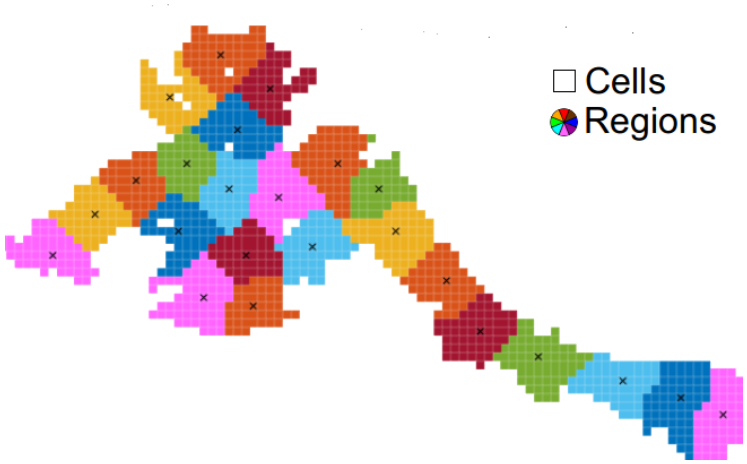
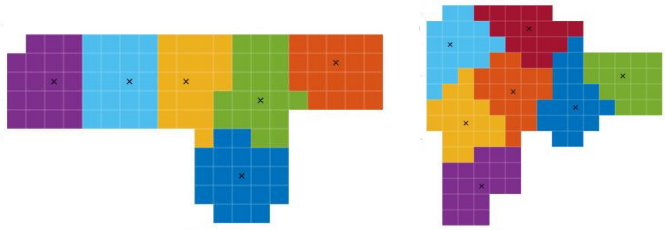


Figure 7: Bin discretisation.

As explained before, the experiments assume that it is not possible to observe the state of the whole environment at once, so instead a set of observable regions for each dataset are defined. During exploration, only the people passing within the boundaries of the chosen region are taken into account to update the model at a given interval in time, while the rest of the environment remains unseen. The division is done into 24 regions for the ATC dataset, 6 for the corridor, and 7 for the office environment, giving an average of 52, 19, and 18 observable m^2 per region, respectively. The 3D lidar human detector used for the creation of datasets achieves good performance up to around 10 meters, which translates to an area of roughly 300 m^2. However, this is in an ideal scenario with open space and no occlusions, which is usually not the case. Therefore, for the ATC dataset it has been chosen to have full 360-degree coverage with a down-scaled radius of 4 m, corresponding to an area of ~50 m^2. Using this same 50 m^2 for the smaller corridor and office datasets would mean having only 2 regions, so to make it more interesting instead an area was chosen in the order of the typical coverage of a depth camera (6 meters range and 60 degrees of horizontal field of view, corresponding to ~19 m^2). In order to partition each environment into spatial regions, the k-means algorithm [10] is used, optimising such that all regions should contain the same or a very similar number of active cells. The summary of the parameters used in each data set can be seen in Table 1 and the environmental area division in Figure 8.



(a) ATC



(b) Corridor (c) Office

Figure 8: Spatial division in observable regions for each environment map.

Regarding time, the same parameters are used for all three datasets. The interval for creating the histograms employed as the input for the STeF-map model creation is set to 10 min. The same interval is used to provide a single observation, i.e. every 10 minutes the exploration strategy can decide whether the robot should stay in the same region or instead move to a different one. If the robot has to move to another regions, the path from the centroid (marked with crosses in Figure 8) of the current region to the centroid of the goal region is computed by means of the A* algorithm ([11]). During the travelling phase, a constant robot speed of 1 m/s is assumed (also assuming that there is no interaction with the people moving around), and that the robot is only able to see what happens inside the region currently being traversed at each instant of time.

For each environment, the total available time for exploring in each day used is 12h (the active hours), corresponding to $\tau = 72$ time intervals (10 minutes each). For the ATC dataset the starting time is 09:00, and for the corridor and office the starting time is 08:00. The rest of the time, the environment is inactive/empty and all the cells in the environment are set to 0. From these 72 time intervals available, data gathering only happens in a certain number of time intervals, which is proportional to the exploration ratio set in each case (e.g. if the ratio is 25%, only 18 of the 72 intervals are used for exploration, while the rest remain unused).

The recalculation of the entropies for both the temporal and spatial domains are done at the end of each explored day, also at midnight the spatial-temporal schedule for the following day is created.

Every cell in the map can present different periodicities corresponding to different human activity patterns. So, in the experiments, each cell's entropy is calculated with either

Dataset	Train	Validate	Test	Regions	Region size	Cell size	Cells
ATC	42 days	2 days	2 days	24	~ 52 m^2	1×1 m	1248
Corridor	10 days	2 days	2 days	6	~ 19.5 m^2	1×1 m	117
Office	18 days	2 days	2 days	7	~ 18 m^2	1×1 m	126

Table 1: Summary of the spatial and temporal parameters used in each data set

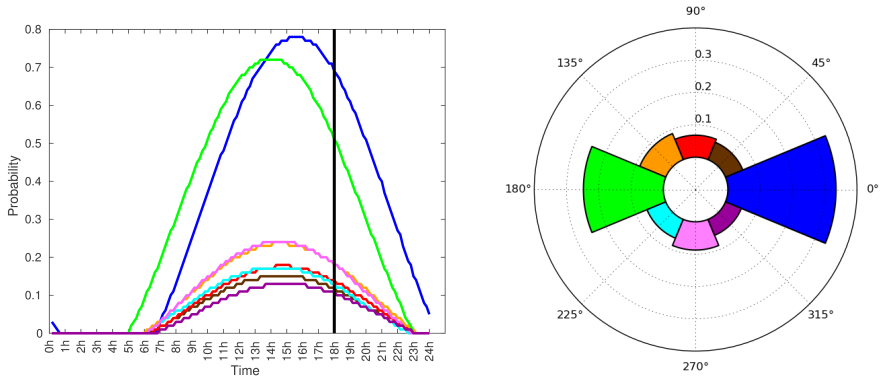


Figure 9: Model prediction over 24h with m=1, and probability distribution of each orientation at t = 18:00.

1 or 2 periodicities, which is usually enough to represent the environment dynamics [12]. Using the validation days, the best number of spectral components for each cell is chosen. These are used later to compute the model predictions and loss in model quality over the testing days.

2.3.4 Evaluation metric

In order to compare the performance of the different exploration strategies, a metric is needed that evaluates the prediction quality from the models. The output of the trained spectral models provides a function for each orientation in each cell. So, for each time t, we obtain a normalised distribution, describing how probable it is to find a person moving in each direction. For example, Fig. 9 presents the prediction graph for a single cell in the map (using $m = 1$) over 24 hours after some days of training. In this cell, there are 2 dominant orientations, one with higher probabilities in the morning and the other in the afternoon. At t = 18:00, the distribution of the 8 normalised orientation probabilities is shown in the polar histogram on the right.

However, obtaining the same orientation distribution with real data at a single time instance t is not possible, because we cannot count sufficient detections to build a meaningful distribution. Instead, the proposed idea is to compare the distribution obtained with the predictions against the ground truth during a defined interval of time. Then, assuming both prediction and ground truth histograms are normalised, the Chi-squared [χ^2] distance is used to indicates the level of similarity between the predicted and ground truth discrete human motion distributions. The higher the distance, the less accurate is the model prediction compared to the ground truth. The total distance of a whole map for a single interval can be defined as

$$\chi^2_{\text{map}} = \sum_{c=1}^{n} \Big(\sum_{b=1}^{k} \frac{(x_b - y_b)^2}{(x_b + y_b)} \Big), \tag{8}$$

where n is the number of cells, k is the number of angular bins for the direction of people motion in the cells, x_b is the value of bin b in the predicted orientation histogram, and y_b is the value of the same bin b obtained from the ground truth data.

Since the χ^2 distance is not a very intuitive measure as it has no units, in the results section the prediction accuracy is expressed as the percentage loss in model quality. The loss is based on how much worse the model prediction is compared to the distance obtained with a model created with full observability in time and space, i.e. with a 100% exploration ratio and always seeing all the cells in the map. The closer the value to 0, the better the predicted model and hence, the better the performance of the spatio-temporal strategy.

2.4 Results

In this section, the result obtained for the 3 different datasets are presented (Figs. 10 to 12). For each one, the 9 possible spatio-temporal exploration combinations are tested (R-R, B-R, E-R, R-B, B-B, E-B, R-E, B-E and E-E), with 3 different exploration ratios: 50%, 25% and 12.5%.

Since the *Random* and *Entropy* strategies produce stochastic policies, the values obtained are shown using a boxplot over 10 runs (median in yellow, interquartile in green, minimum and maximum with the dashed lines, and potential outliers in red). Even though *Round-Robin* produces a deterministic policy for visiting the areas/intervals, the results are also expressed with confidence intervals, since it has been defined that the starting region is not always the same. The values obtained are always computed at the end of the total training days corresponding to each dataset (Table 1). The validation days are used to decide the number of periodicities for each cell, and the testing days are used to obtain the percentage loss in model quality.

Furthermore, for the three "pure" combinations possible, namely R-R, B-B and E-E, the model quality loss at certain days during the total exploration days is computed, always using the same validation/training days (at the bottom of each Fig. 10, 11 and 12). As before, 10 runs per strategy are computed, but in this case plotting the results as the average together with a 95% confidence interval, assuming Student's t-distribution.

2.4.1 Per-dataset observations

ATC dataset

The results obtained for the ATC dataset are summarised in Figure 10. As a general overview, we see that the strategies with an entropy-based policy perform consistently better than their uninformed counterparts.

For low exploration ratios, as in the 12.5% case, we found that E-R, E-B and E-E produce better results, showing that taking into account the entropy in the spatial domain is a key factor. However, as the available time to explore is increased, the strategies which take into account the entropy to decide when to explore (R-E, B-E, E-E) obtain better results, and the spatial domain becomes a secondary factor.

Observing the temporal evolution, we see that B-B exploration obtains by far the worst results, being the slowest one and only being able to catch up with the rest of the strategies in the period from day 35 to 42. For the 50% case, R-R and E-E perform very similarly, but for 25% and 12.5% after day 25, approximately, E-E obtains slightly but significantly better results.

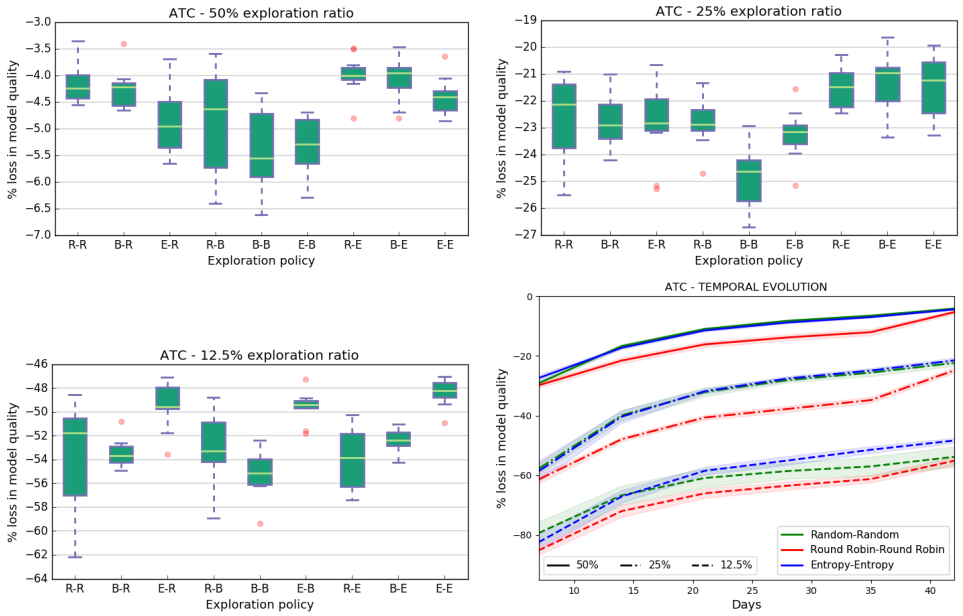


Figure 10: Results for the ATC dataset with 50, 25 and 12.5% exploration ratios with the 9 spatio-temporal exploration strategy combinations, and temporal evolution over the exploration days for the 3 pure combinations.

Corridor and Office datasets

For the last two datasets (Figs. 11 and 12), the results are a bit more difficult to analyse, due to the fact that the deviations on the results tend to be much bigger compared with the ATC dataset. We believe this is a consequence of having a much lower number of people occurrences in each day. So, sometimes, seeing a person in a certain interval of time becomes just a matter of luck.

Nevertheless, we found that for the 25% exploration ratio in the Corridor dataset, the B-B exploration strategy obtains consistently better results, which is also confirmed by observing the temporal evolution. However, for the 50% and 12.5% exploration ratios, we cannot extract any meaningful conclusions, as all of them behave similarly and there are no patterns on either the spatial or temporal side.

For the Office dataset, we found that the exploration strategies which follow the *Round-Robin* policy in the temporal domain perform worse for low exploration ratios, which can also be seen in the temporal evolution plot. The major difference appreciated in this dataset comes for the 25% ratio, where the combinations sharing the entropy-based exploration in the time domain (R-E, B-E and E-E) manage to obtain consistently better results.

2.4.2 Discussion

The results suggest that the entropy-based exploration works well when we have an environment with a substantial number of people detections and somewhat regular flows, as in the ATC dataset. In scenarios with lower human encounters, like the Corridor and Office environments, it would probably be necessary to further extend the days explored to several weeks to obtain statistically significantly results, or compute hundreds of runs to deal with the higher deviations obtained. Also, in the Corridor and Office environments

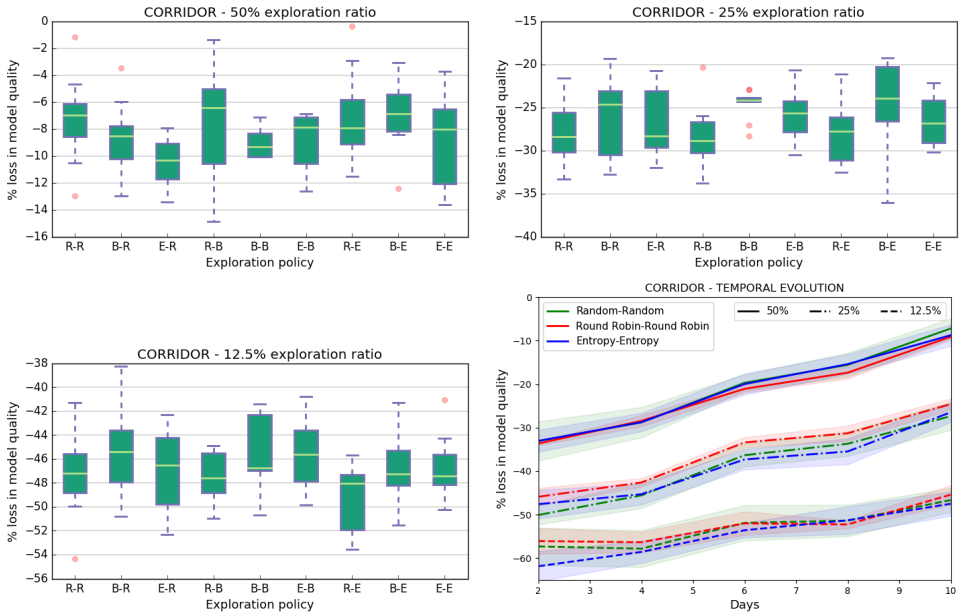


Figure 11: Results for the Corridor dataset with 50, 25 and 12.5% exploration ratios with the 9 spatio-temporal exploration strategy combinations, and temporal evolution over the exploration days for the 3 pure combinations.

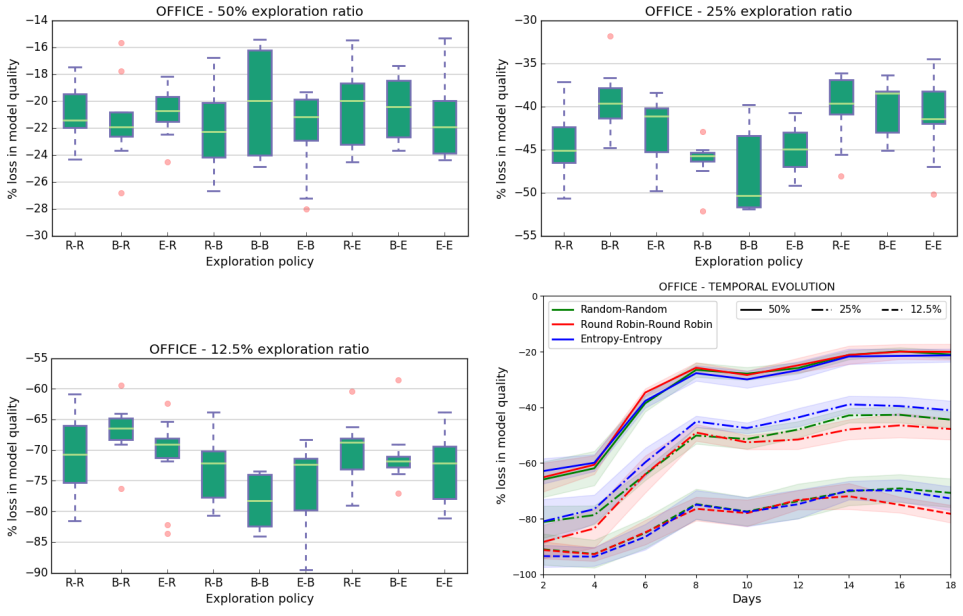


Figure 12: Results for the Office dataset with 50, 25 and 12.5% exploration ratios with the 9 spatio-temporal exploration strategy combinations, and temporal evolution over the exploration days for the 3 pure combinations.

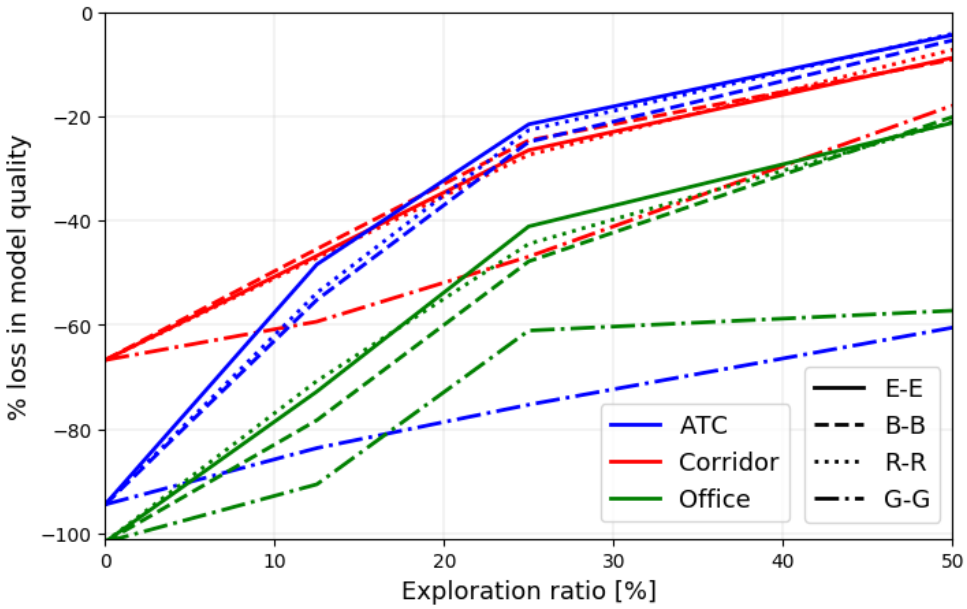


Figure 13: Correlation between model quality loss and exploration ratio for all datasets with the 3 pure exploration strategy combinations (E-E, B-B, R-R).

at any point the exploration strategy in charge of scheduling the region to explore has a major impact on the final results. Probably the fact that there are a much lower number of regions to explore compared to the shopping centre makes no actual difference, as the simulated robot is able to visit them a lot more times, even for low exploration ratios.

However, taking the exploration ratio as clearly the factor that has the biggest impact on the total percentage loss in model quality during the exploration activities (which is clear to see in the temporal evolution plot in Figs. 10 to 12), we see that in all datasets this impact follows a similar trend. In Fig. 13 the correlation between the percentage loss in model quality and the exploration ratio is plotted for all three datasets, at the end of the corresponding exploration days for the three pure combinations (E-E, B-B, R-R). The value obtained with 0% exploration ratio corresponds to a model which has not been trained with any data, so all orientations in each cell have the same probability. The outcome shows that the loss tends to decrease exponentially as the exploration ratio is increased.

Looking at the loss in model quality also in Fig. 13, the Office dataset obtains noticeably worse results for all exploration ratios compared to the Corridor and ATC datasets. We think this is caused by the fact that the recordings in the Office dataset were also done during the weekends, days which present a very different behaviour (the environment is mostly empty) when compared to the weekdays. This increased complexity makes it more difficult for the spatio-temporal model to find the overall patterns to be modelled using only sparse partial observations. In the ATC dataset, weekends are also part of the data, but in this case, there are no major differences in human motion behaviour between the weekday and weekend recorded.

2.5 Summary

In this report, a comparison between multiple robot exploration strategies to build a spatio-temporal model of human motion in a given environment is presented. Moreover,

it is proposed to use the data already gathered by the robot to determine where and when to perform future observations, based on the entropy levels in the model, which are computed from the distributions of pedestrian motion direction.

The results show that the entropy-driven policy improves the results obtained by the uninformed exploration strategies in scenarios with a substantial degree of human presence and rhythmic patterns of activity. On this issue, future work will aim to do a more in-depth analysis to quantify the characteristics of a given environment and analyse the differences.

Furthermore, it has been demonstrated that the exploration ratio is the key factor affecting the model prediction quality and that similar trends in the correlation between model quality loss and the exploration ratio are obtained. This is interesting, considering that all three datasets tested in the experiments contain a different number of exploration days and a different number of regions to explore.

At the time of writing, unfortunately, it has not been possible so far to test the reported methods in a real warehouse scenario as planned, due to the current Covid-19 lockdown restrictions. Ongoing work therefore includes development of a realistic warehouse simulation, based on existing tools developed so far in the ILIAD project, to enable us to evaluate and compare the performance of the respective methods under more realistic operating conditions. We will also endeavour to carry out further experiments at one of the industry partner premises, if and when allowed under the Covid-19 restrictions.

References

[1] Sergi Molina, Grzegorz Cielniak, Tomáš Krajník, and Tom Duckett. Modelling and predicting rhythmic flow patterns in dynamic environments. In *TAROS*, pages 135–146, 2018.

[2] Tomasz Piotr Kucner, Martin Magnusson, Erik Schaffernicht, Victor Hernandez Bennetts, and Achim J. Lilienthal. Enabling flow awareness for mobile robots in partially observable environments. *IEEE Robotics and Automation Letters*, 2(2):1093–1100, April 2017.

[3] Sergi Molina, Grzegorz Cielniak, and Tom Duckett. Robotic exploration for learning human motion patterns. In *submitted to Transaction on Robotics (TRO)*. IEEE, 2020.

[4] Sergi Molina, Grzegorz Cielniak, and Tom Duckett. Go with the flow: Exploration and mapping of pedestrian flow patterns from partial observations. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 9725–9731. IEEE, 2019.

[5] George A Miller and William G Madow. On the maximum likelihood estimate of the shannon-wiener measure of information. *Readings in mathematical psychology*, 1:448–469, 1963.

[6] Liam Paninski. Estimation of entropy and mutual information. *Neural computation*, 15(6):1191–1253, 2003.

[7] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[8] D. Brscic, T. Kanda, T. Ikeda, and T. T. Miyashita. Person position and body direction tracking in large public spaces using 3d range sensors. *IEEE Transactions on Human-Machine Systems*, 43(6):522–534, 2013.

[9] Zhi Yan, Tom Duckett, Nicola Bellotto, et al. Online learning for human classification in 3d lidar-based tracking. In *International Conference on Intelligent Robots and Systems (IROS)*, 2017.

[10] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982.

[11] Peter E Hart, Nils J Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

[12] Tomáš Krajník, Jaime Pulido Fentanes, João Santos, and Tom Duckett. FreMEn: Frequency map enhancement for long-term mobile robot autonomy in changing environments. *IEEE Transactions on Robotics*, 2017.