Human Exploration Patterns in Unknown, Time-sensitive Environments

Phillip M. Walker, Andreas Kolling, and Michael Lewis
School of Information Sciences
University of Pittsburgh
Pittsburgh, Pennsylvania, USA
{pmw19@pitt.edu, andreas.kolling@gmail.com, ml@sis.pitt.edu}

***Abstract*—Exploration of unknown environments has numerous applications in the domains of search and rescue, pursuit-evasion, map building, and military espionage and intelligence gathering. While it is ideal to have autonomous robots search intelligently based on global patterns and features of each environment, such learning algorithms should also be informed by, and perform at least as well as, human exploration. Most autonomous exploration algorithms focus on local and greedy choices and are less concerned with learning to recognize structural features of environments in order to improve global exploration performance. If the strategies people use during exploration can be determined, they can be used as a basis for creating good utility functions for autonomous robots to be optimized by machine learning techniques. Here, we perform a study where human participants explore both random and patterned environments, in two different time-sensitive scenarios, in an attempt to determine what strategies people use to maximize area explored under a variety of cases. We found that participants in a controlled study did in fact adapt their exploration strategies in time-sensitive scenarios and also exploited features of the environments in order to achieve better global exploration performance.**

***Index Terms*—Human-Robot Interaction, autonomous exploration, human exploration.**

I. Introduction

In the past few decades, significant progress has been achieved in the area of autonomous robotic exploration, mapping, and target searching. This progress is essential to improving and expanding on numerous types of tasks and domains for mobile robots, such as exploring burning or collapsed buildings and mining shafts for survivors and safe pathways, to exploring hostile territory on the battlefield. Because both of these situations pose significant hazards to the health or life of the human team involved, such exploration is an optimal scenario for autonomous robotic teams.

The primary issue facing autonomous robotic exploration of unknown environments, however, is how to explore in the quickest and most efficient manner possible for the given scenario. For instance, if the value of new explored area decreases quickly, it may be advantageous for a robot to explore larger areas first to gain as much new information as possible, and then double back to fill in missed areas later. However, if the value of information decays much more slowly, one might want a robot to be more thorough at the beginning so as to avoid having to waste time backtracking at the end. This concern is important, as both search and rescue and intelligence gathering involve deadlines, after which the value of the information gathered by exploration decreases or disappears altogether. For instance, knowing about the location of victims trapped inside a smoke-filled building is much more valuable in the first few minutes than hours later. Achieving autonomous exploration that can take into account environmental features, patterns, and time constraints is essential if we are to have teams of autonomous robots perform in a manner on par with human exploration.

Clearly, there are significant advantages to having robots replace humans in dangerous exploration tasks, and researchers have tackled this goal in several ways. For instance, [1] demonstrate an algorithm employing a utility function that looks several steps ahead when evaluating potential sensing locations at each step. The algorithm evaluates each location based on criteria such as the number of corners visible from the candidate location, orientation change required to reach the next location, and length of the shortest edge to the candidate location. Similarly, [2] focus on the goal of maximizing exploration performance when only short-term, local information is available to the simulated robot. Their algorithm combines a number of features–both known and estimated–into a utility function employing a Choquet integral, and selects the best candidate location based on the resulting optimization over all of the options.

Exploiting environmental features to extract patterns and qualitative information, and thereby improve exploration, is also a significant focus of this paper. To this end, [3] attempt to use environmental features to help each robot determine what type of location it is in during a search and rescue scenario–for instance, whether or not the victim just found is located in a room or the middle of a hallway. This method could also be used to help a robot make decisions about candidate pathways for future exploration. For example, if the robot is able to determine that it is in a room, it might be able to make an estimate of how large the room is based on others it has seen in past exploration of the environment. Most importantly, this hypothesis is backed up by the results shown in [3], which demonstrate that semantic place information can indeed speed up exploration and time needed to locate previously unknown targets.

Ultimately, this work should be extended and fully implemented in teams of several robots, rather than just one. This has obvious advantages, as the amount of area that can theoretically be explored by multiple robots in a given amount of time is much larger than a single robot, and it also prevents the case where exploration halts altogether due to internal or external damage to the robot's movement or sensing capabilities. However, with the increased robustness and exploration capabilities of multiple robots comes the challenge of coordinating them. While there are numerous problems being addressed in multi-robot exploration, such as localization and odometry error correction (see [4], [5], and [6]) and bandwidth constraints ([7]), the task of efficiency is one of the more basic and theoretical hurdles to be overcome. To this end, researchers have begun to explore techniques to maximize the utility achievable from multiple robots performing the same exploration task by preventing overlap as much as possible. The method employed by [8] attempted to solve this problem with considerable success in both simulation and real world environments by lowering the utility of unseen area around an exploration point after it had already been selected by one of the robots in the team. Furthermore, their algorithm is also applicable to many types of exploration and environment representations, such as the coverage maps employed by [9]. Conversely, [10] approach the problem of multi-robot coordination by transforming it into a version of the multiple traveling salesman problem (MTSP) that employs an auction-based task allocation method. Furthermore, as the MTSP is *NP-hard*, their overall algorithm is modified to take an incremental approach that allows for changing objectives or dynamic tasks.

While there is extensive research in the domain of autonomous exploration and mapping, there is significantly less on how humans that manually guide robots explore unknown environments, or how they interact with robots that explore semi-autonomously. This paper will mainly focus on the former problem–determining which strategies people use while guiding an exploring robot, and what environmental patterns they exploit. Ultimately, we hope to apply the results to autonomous robots to see if they can improve their own, initially blind, learned strategies. If human explorers are found to use certain semantic or qualitative features about the environment to improve exploration, such abilities should be built into exploring robots as well.

There has been considerably work looking at how humans store representations of their environments during exploration tasks. Cognitive maps (see [11] for an overview) are used to describe how humans maintain spacial and qualitative information about the layout of their surrounding environment. Researchers, such as [12], and as far back as [13], have argued for cognitive maps in other animals as well; [14] even argued for cognitive maps in honey bees. Some, however, have argued that the traditional explanation of cognitive maps is flawed. For instance, [15] argue that the standard definition of cognitive maps is too restricted, and argues for more robust models, such as cognitive collages and spacial mental models, that can be applied to a wider range of scenarios.

Applying this work to robotic exploration, the work in [16] attempts to use cognitive maps for mobile robots to give them a better understanding of the surrounding environment. Their method allows autonomous robots to classify, and later identify, different places in the environment by using information inherent in the layout and nearby objects. In an attempt to utilize human knowledge of the environment to guide robot exploration, [17] developed a system whereby participants could control a robot by spoken commands–thus allowing them to use semantic information that cannot be identified or used by the exploring robot to determine which paths to take through the environment. Another problem that could be addressed by combining research in human and autonomous robotic exploration is the exploration vs. exploitation dilemma. This concerns that fact that, in nearly every exploration task, the exploring agent must decide whether to continue using its possibly suboptimal strategy to explore, or whether it should make a few random, or even poor, choices in hopes of discovering a significantly better strategy. Significant research has been performed addressing this problem (see [18] for a survey of the exploration vs. exploitation problem in human participants).

The goal of this paper is to test our hypothesis that participants who have the best exploration times and scores will use different environmental features depending both on the type of map and type of decay function in the given condition. In other words, the successful participants will have noticeably different strategies than the others. In the future, we hope to use these results to see if simulated autonomous robots using similar strategies and feature weights will perform better than robots with other exploration strategies.

II. Problem Description

We are concerned with the problem of human-guided robotic exploration of unknown environments, when the value of new information about the environment decreases predictably over time. The environment is modeled by a workspace $\mathcal{W} = \mathbb{R}^2$ (or $\mathbb{R}^3$) with obstacles $\mathcal{O} \subset W$ and free space $\mathcal{F} = \mathcal{W} \setminus \mathcal{O}$. We suppose that $\mathcal{F}$ is connected. A robot trajectory is given by $\pi : [0, T] \to \mathcal{F}$ with time horizon $T \in \mathbb{R}^+$. For simplicity, in this paper we are considering only one robot; extensions to multiple robots will be addressed in future work. At a location $\pi(t) \in \mathcal{F}$, the sensor footprint of a robot is $S(\pi(t)) \subset \bar{\mathcal{F}}$ (closure of free space). For our examples, we assume a circular sensor footprint with range r, but in principle any footprint is admissible.

An exploration strategy $\mathcal{S}$ for a robot is a sequence of exploration locations $\mathcal{S} = l_1, \ldots, l_n$ that the robot visits. While these locations are effectively discretizing the trajectory we set for a robot, it is a common approach to have a robot select a location from a set of frontier points at the boundary between visited and unknown space. Let $E(t) \subset \mathcal{F}$ be the explored region while $\mathcal{F} \setminus E(t)$ is the *unknown* region. Initially, all of $\mathcal{F}$ is *unknown*. In most scenarios no prior map of $\mathcal{F}$ will be given, but our approach applies for scenarios with and without prior maps.

When a robot reaches an exploration point, it adds to its score according to utility of the area discovered at that location. To compute this utility let $d : [0, T] \to [0, 1]$ be a monotonously decreasing function with $d(0) = 1$ and $d(T) = 0$ denoted as *decay* function. A robot arriving at l_i at time t_i receives a utility $u(l_i) = d(t_i) \cdot \int_{x \in S(l_i) \setminus E(t_{i-1})} 1 \, dx$.

In a simple coverage problem we would have a constant $d(t) = 1, \forall t \in [0, T]$. In a disaster scenario, however, where the environment is expected to collapse around some time in $[t_{critical}, T]$ we would have the decay function decline dramatically prior to $t_{critical}$ and then slowly approach 0 until T. The total utility a robot receives for a strategy $\mathcal{S}$ is now:

$$U(\mathcal{S}) = \sum_{l_i \in \mathcal{S}} u(l_i).$$

One may also consider adding a cost for distance to $u(l_i)$. But with a strictly decreasing $d(t)$ a cost for distances is given implicitly as longer travel times incur a faster decay of utility.

If $\mathcal{O}$ is known, one could attempt to compute an optimal exploration strategy $\mathcal{S}$ that maximizes $U(\mathcal{S})$ (with or without added costs for distanced travelled). Obviously, this problem is expected to be hard, and becomes even harder when extending to multiple robots (in which case even the assignment of exploration locations is *NP-hard*). In most scenarios, however, $\mathcal{O}$ is unknown. In these cases, where exploration is done autonomously, much of the prior work resorts to greedy approaches. These usually involve some set of criteria, computed for every candidate location, that are combined to select the best one to be visited next. One of the downsides of such greedy approaches is that they do not take into account the effect of a choice onto subsequent choices; if the environment exhibits a type of structure or pattern that can be exploited, then such approaches fail to utilize this.

Our goal is to determine if participants can take into account the time constraints and structure of the environment to learn the best way to exploit patterns that improve the overall exploration score, when the choice of the next exploration location still only relies on local information. To illustrate this idea consider the following example. A participant is guiding a robot through an environment with two types of exploration locations–those at doors to small rooms and those leading through corridors and intersections of corridors laid out in a simple grid with all corridors either horizontal or vertical as seen in Fig. 1. Given a fast decay the participant should prefer to explore hallway locations as they increase the area explored fastest without having to enter the small rooms. However, with a slow decay and sufficient time T to explore most or all of the environment, it would be better to explore each room as the robot passes to avoid having to travel long distances back once all corridors are explored. Hence, a particular type of exploration strategy can evolve (one that can theoretically be learned by an autonomous robot as well) out of the relationship between the structure of the environment and the criteria at exploration locations.

III. Experimental Design

To evaluate our hypothesis that participants would explore an environment with embedded global patterns in a systematic

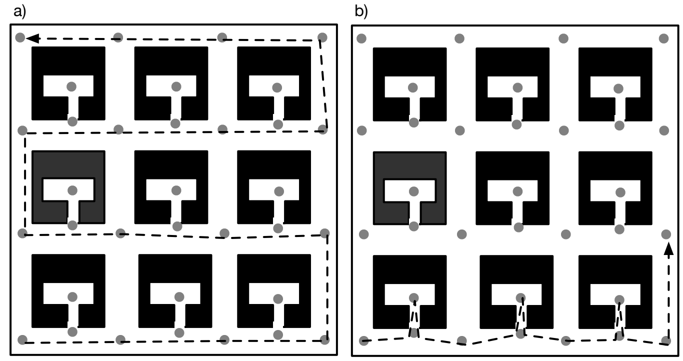


Fig. 1. An illustration of two types of exploration strategies. Free space is white and exploration locations the robots can choose are marked in grey. Paths are shown with a dashed line. In a) the robot prefers to move between hallways while in b) it prefers to clear rooms. The robot in a) will have to travel through an overall longer distance to explore the entire environment while the robot in b) chooses the optimal path in terms of distance for exploring all of the environment but may received a lower utility if $d(t)$ decays fast.

way, a study was designed that compared the exploration patterns used in an office building-type environment with perfect symmetry over the x- and y-axis with environments randomly populated with obstacles. The office building layouts were designed in such a way that both global and local patterns existed that could be found and exploited by the participants to gain maximum coverage, whereas the random forests contained rectangles placed semi-randomly in an otherwise open environment. In both types of environments two different information value decay conditions were used–linear and exponential.

A. Decay Functions

The two decay functions served as a way of discounting the score gained by new explored area as more time went by during the experiment. In the first function, the discounting multiplier was defined by following linear function, down to a minimum value of 0.5:

$$d_{lin}(t) := \max\left\{1 - \frac{t}{T}, 0.5\right\}. \tag{1}$$

Here, t represents the total time passed since the beginning of the trial, and T is the maximum time given for the trial, which was 240s for each environment.

In the second condition, points were discounted by a multiplier defined by the following exponential function, down to a minimum of 0.1:

$$d_{exp}(t) := \max\left\{1 - \frac{1}{1 + e^{-z(t)}}, 0.1\right\}, \tag{2}$$

with $z(t)$ given by $z(t) = \frac{t}{T} \cdot 16 - 8$. Note that $z(t)$ can be readily modified to change the time at which its value decreases dramatically.

Fig. 2 shows the behaviors of these two decay functions–with $d_{exp}(t)$ allowing for greater gains in the early stages, but decaying much more quickly, and settling at a lower minimum, than $d_{lin}(t)$.

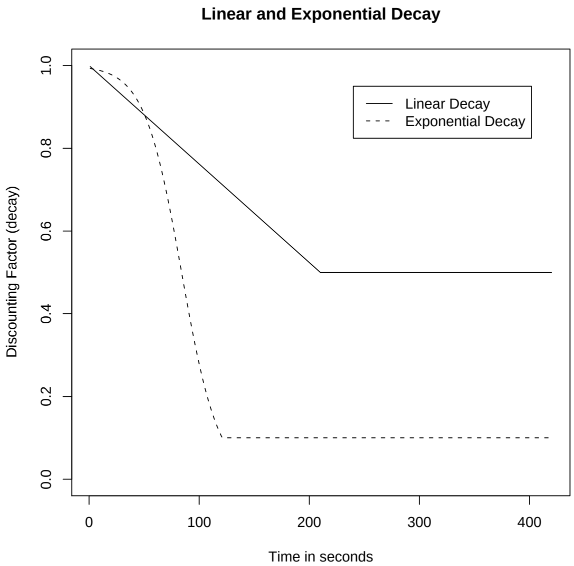


Fig. 2. A plot of the linear and exponential decay functions.

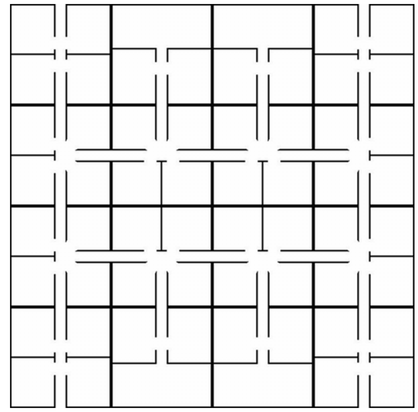


Fig. 3. An example of a patterned floor plan with perfect symmetry.

Fig. 4. An example of a random environment.

B. Environment Setup

We consider two types of environments, random and patterned (i.e. the office environments). Both environments were designed using sets of predetermined "blocks" that were then pieced together using the environment population algorithm described below. This allowed the environments to remain relatively consistent, and, in the case of patterned environments, allowed for the construction of global patterns such as hallway loops and symmetries. Each block was 200×200 pixels wide, and contained sensing locations semi-uniformly dispersed throughout the open areas of the block. For the patterned environments, seventeen types of blocks were used, ranging from completely open or completely closed to more complicated blocks. The random environment consisted of thirteen types of blocks, which belonged to either a low, medium, or high density class (dictating the number of obstacles within the block). Otherwise, the obstacles in these blocks were placed randomly, with the exception that any openings between obstacles had to be large enough for the simulated robot to move through.

The blocks in the patterned environments were pieced together so that they simulated real-life office building floor plans as closely as possible. Therefore, the blocks were initially assigned specific probability weights–reflecting how often they would be chosen by the population algorithm–that were updated based on surrounding blocks. This ensured that there were no arbitrary dead ends or too many isolated sections, and that each environment included hallway cycles, a mixture of open areas, and hallways of varying lengths with small office rooms coming off of them.

To create the patterned environments, the population algorithm began by dividing the environment into four quadrants, and populating the top left quadrant by randomly sampling blocks from the block pool according to their current probabilities. Then, this quadrant was reflected over the x-axis, and then the entire left side was reflected again over the y-axis to fully populate the environment (Fig. 1). The random environments were created by giving each random block and equal probability, and then populating the environment from left to right, and top to bottom (Fig. 2). Each environment was a 4×4 grid of blocks, resulting in a 800×800 pixel image of the environment.

The consideration of environments created by some underlying generator is necessary since we do not want to overfit, i.e. by chance exploit patterns of a particular instance of an environment, but rather the exploit general pattern that all environments of a type share.

C. Methods

Twelve participants from the University of Pittsburgh community participated in this study, and were compensated for the experiment's one-hour duration.

The participants were told that the study was looking at exploration of unknown environments, and that their goal was to maximize their total score. They were informed that their score depended on how much new area was explored at each point, and also on the current pixel value of that trial's decay function at that point. The two decay functions were then described to participants, and they performed two training trials–each of them a random environment with one of the two possible decay conditions–to get a feel for the time constraints in each condition. Each training trial lasted four minutes.

After this initial training, participants explored four random obstacle environments, followed by four patterned, office-like environments. In each block, the decay function for each trial alternated, starting with exponential decay in the random environment trial block, and with linear decay in the patterned trial block. Participants had four minutes for each trial, after which the experiment concluded. The participants were instructed to try an maximize their total score in whatever way possible.

The interface was the same across all participants and environments, and provided participants with information about the current value of the decay function in the form of a bar on the left-hand side that corresponded the the current value of newly-explored area. The interface also displays their cumulative score, time left to explore, score at the last location, and the available exploration locations the robot could move to via the main window. The user could easily scroll across the entire map to see explored and unexplored areas using the right mouse button. Fig. 5 shows a snapshot of the interface.

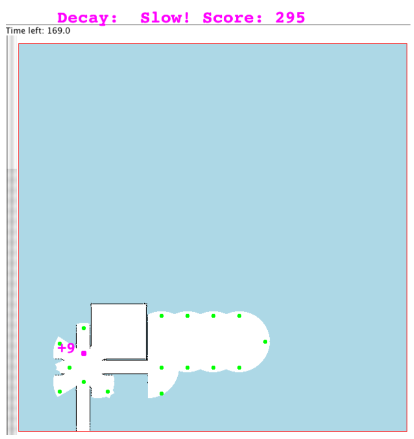


Fig. 5. The interface used in the experiment as seen by a participant. The robot is marked as a larger pink dot while the available exploration locations are marked in smaller green dots. Unknown area is light blue and explored area white. On the left is a bar that indicates the value of the decay function.

IV. Results and Discussion

In our analysis of the results, we attempted to answer questions that would determine whether human participants could recognize patterns embedded in environments and exploit them to improve their exploration strategies. We also wanted to determine if participants would adapt to differing decay functions, and, in essence, sense the varying degrees of urgency and update their strategies accordingly.

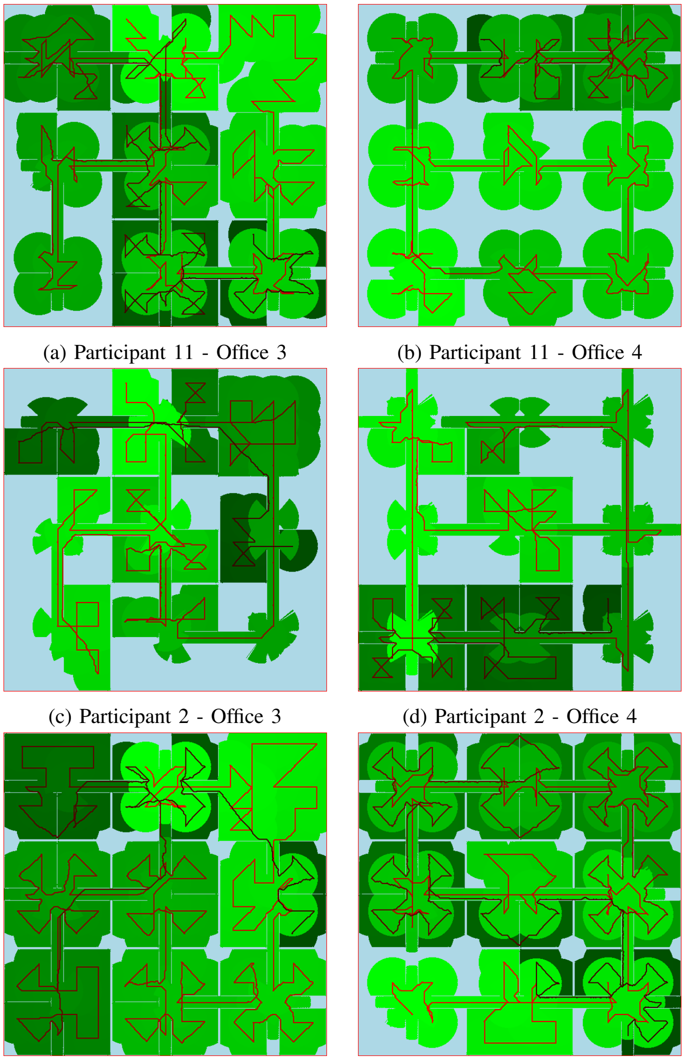
In the following we will compare the performance of all participants with a predefined exploration strategy for the office environments. To allow a comparison this predefined strategy is also executed by a human operator, further denoted as the expert, who is using the same interface as the participants but is instructed to explore the environments as follows. In the linear decay condition the expert is required to clear only larger areas within the first thirty seconds and then proceed by clearing every room fully once entered. After a room is cleared the next closest room is selected. The expert is additionally instructed to minimize the overall travelled distance and to attempt to explore the entire environment. This expert strategy emulates an efficient solution to the problem of finding the shortest path that explores the entire environment[1]. For the exponential decay the expert is instructed to only clear rooms partially by moving only to the entrance node, because in most rooms, more than half the room is already explored after visiting the node by the entrance. Hence, following this strategy a large area can be cleared early on, but rooms will have to be revisited. This expert strategy emulates an efficient strategy to optimize the score for the exponential decay. Fig. 6 parts (e) and (f) show these two types of predefined strategies executed by the expert operator. Four example strategies of participants, one with low scores and one with a high scores, are also shown in Fig. 6 parts (a), (b), (c), and (d). Table I shows the average scores and area covered by participants and the expert. Area is measured in units of 100 square pixels and means are reported with their standard deviation seperated by $\pm$.

TABLE I
Summary of Scores and Area Explored

Map	Decay	Score	Expert	Area	Expert
Random 1	d_{exp}	1610 ± 236		2256 ± 294	
Random 2	d_{lin}	1636 ± 120		2398 ± 145	
Random 3	d_{exp}	1658 ± 187		2551 ± 104	
Random 4	d_{lin}	1724 ± 72		2505 ± 70	
Office 1	d_{lin}	1385 ± 205	1992	2212 ± 331	3064
Office 2	d_{exp}	1368 ± 199	1996	2367 ± 239	2908
Office 3	d_{lin}	1502 ± 197	2021	2329 ± 298	3099
Office 4	d_{exp}	1371 ± 248	2025	2242 ± 240	2836

The total area in 100 unit pixels for office maps 1 to 4 was 3313, 3322, 3318, and 3281, respectively. For random maps 1 to 4 the area was 2752, 2806, 2841, and 2852, respectively. Participants needed on average 1.50 ± 0.28 seconds to choose a new exploration location for the four office maps (on average 1.57, 1.49, 1.47, and 1.47 seconds in office maps 1 to 4, respectively). In random environments participants took 1.99 ± 0.45 seconds to choose a new location. The expert operator needed on average 0.89 seconds for each choice. The total number of locations visited by participants was on average 131 ± 23, 132 ± 19, 131 ± 21, and 126 ± 16 respectively. The expert operator visited 166, 150, 168, and 150 locations. Part of the improved performance of the expert is hence to be attributed to faster decisions. Overall, all office environments and all random environments were very similar with regard to performance and in the following we shall average across all environments of the same type and condition. There were, however, significant individual differences as seen by the large confidence interval plotted in Fig. 7 and exemplified by the two example participants in Fig. 6.

[1]Note that an actual optimal strategy would involve solving a problem similar to a traveling salesman problem even if the map and exploration locations were known in advance.



(a) Participant 11 - Office 3
(b) Participant 11 - Office 4
(c) Participant 2 - Office 3
(d) Participant 2 - Office 4
(e) Expert - Office 3
(e) Expert - Office 4

Fig. 6. Four examples of strategies. Regions that were explored earlier are lighter (green). Paths of the robots are marked in red and earlier positions are also lighter. Parts (a),(c), and (e) show strategies for linear decay while (b),(d), and (f) show strategies for exponential decay. Participant 11 received scores 1784 for (c) and 1836 for (d). Participant 2 received scores 1263 for (c) and 1068 for (d). The expert scored 2021 in (e) and 2025 in (f).

To determine whether participants adapted to the type of decay, we performed an analysis of the average new area explored within the last 30 seconds at any time $t \geq 30$. A multiple linear regression for all office maps with time ($b = -0.008, p < .001$), decay condition ($b = 1.639, p < .001$), and an interaction of the two ($b = -0.0141, p < .001$) on new area revealed that new area differs significantly depending on time and differently so across decay conditions. The same holds for all random maps with time ($b = -.0652, p < .001$), decay condition ($b = -1.149, p < .001$), and an interaction of the two ($b = .0088, p < .001$) all significantly influencing new area. This indicates that in both types of maps participants adapted their strategies to the decay conditions. Fig. 8 and Table II show the results of a linear regression with time on new area for each condition and each type of map, as well

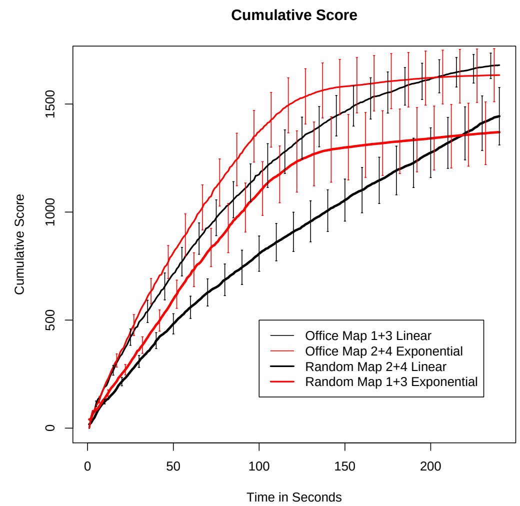


Fig. 7. The cumulative score for both types of environments and decay conditions. The verticals bars indicate the 50% confidence interval ($\pm .6745 \cdot$ standard deviation).

as for the predefined expert strategies. In the exponential decay condition in office environments, participants successfully adapted their strategy to explore more new area early, thereby sacrificing future exploration speed due to increased travel distances. The most extreme example of this adaption is the expert strategy with the largest difference between new area in linear and exponential conditions. The expert strategy serves as an extreme example of adaption to the decay conditions with the largest differences between the regression coefficients. Regarding the interaction between decay condition and time found for forest environments, Fig. 8 shows that this interaction is weaker and inverse of that in the office environments.

One final question is whether the adaption observed above is in fact beneficial, i.e. whether it improves the score. To answer this question, we looked at the score that a strategy would obtain if it were executed with a different decay function. For this we simulated an exploration trial using the exact same choices of exploration locations but instead of the original decay function apply the other. For the expert strategy the final scores drop from an average of 2007 (linear) and 2010 (exponential) for office environments to 1765 (exponential) and 1956 (linear), respectively. For the participants the final scores drop from 1443 to 1257 when applying the exponential decay to the linear decay strategy. For the exponential decay strategy, however, the opposite is observed and the original score of 1370 improves to 1497 when applying linear decay. Note that linear decay assigns a larger value to explored area for most times $t \in [1, T]$ as seen in Fig. 2 which can explain this effect. But it also indicates that participants do not adapt as well as the expert, and would also have fared better in the linear decay condition if they followed their strategy that they employed in the exponential decay condition.

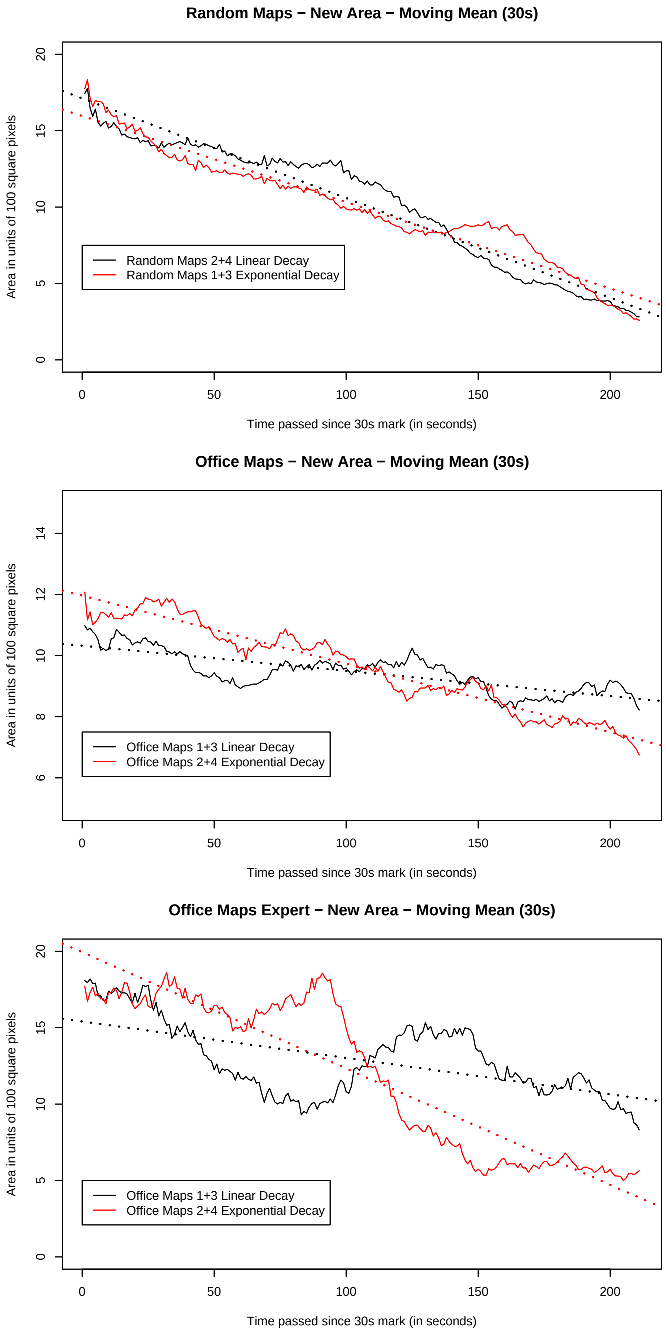


Fig. 8. New area explored within previous 30 seconds for random maps, office maps, and the expert strategy. Dashed straight lines plot the result of the regression analysis from Table II.

V. Conclusion

We showed that human operators can adapt successfully to different types of exploration and can incorporate information about the time-sensitivity into their strategies. Particularly for exponential decay in the value of information, participants performed better. But more importantly, they utilized the environment structure for this adaptation. In office environments, this provided a significant improvement for the final scores. Yet, there were significant differences amongst participants, and only those with the highest scores performed comparable to our predefined expert strategy. These results demonstrate that global structure can be utilized to improve exploration and that human operators, some better than others, can learn to use this structure to improve their performance.

While there are a number of limitations to our study, our main hypothesis has been confirmed. This encourages further work and provides a basis for learning an association between environment's features and successful strategies executed by humans in more complex environments. The computation of environment features has already received considerable attention in the robotics literature and this body of work can to be applied to this problem. Obviously, for this we must consider more types of environments and also those that share many structural features while being distinct in others. The challenge will then be to determine the right type of strategy given ambiguity about the current environment. Ultimately, an extension of this approach to teams of multiple robots and three dimensional environments is envisioned.

TABLE II
New Area - Regression Analysis With Time

Map	Decay	b	a	R^2
Random 1+3	d_{exp}	$-.056$ $(t(209) = -61.47)$	15.96	.95
Random 2+4	d_{lin}	$-.065$ $(t(209) = -60.31)$	17.11	.95
Office 1+3	d_{lin}	$-.008$ $(t(209) = -10.67)$	10.32	.60
Office 2+4	d_{exp}	$-.022$ $(t(209) = -60.07)$	11.96	.94
E. Office 1+3	d_{lin}	$-.024$ $(t(209) = -10.67)$	15.41	.34
E. Office 2+4	d_{exp}	$-.076$ $(t(209) = -34.70)$	19.94	.85
All linear coefficients have $p < .001$; E. = Expert; R^2=adjusted R^2				

Acknowledgment

This research has been sponsored in part by AFOSR FA955008-10356 and ONR Grant N0001409-10680.

References

[1] B. Tovar, L. Munoz-Gómez, R. Murrieta-Cid, M. Alencastre-Miranda, R. Monroy, and S. Hutchinson, "Planning exploration strategies for simultaneous localization and mapping," *Robotics and Autonomous Systems*, vol. 54, no. 4, pp. 314–331, 2006.

[2] N. Basilico and F. Amigoni, "Exploration strategies based on multi-criteria decision making for an autonomous mobile robot," in *Proceedings of the European Conference on Mobile Robots*, 2009, pp. 259–264.

[3] C. Stachniss, O. Mozos, and W. Burgard, "Speeding-up multi-robot exploration by considering semantic place information," in *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*. IEEE, 2006, pp. 1692–1697.

[4] I. Rekleitis, G. Dudek, and E. Milios, "Multi-robot exploration of an unknown environment, efficiently reducing the odometry error," in *International Joint Conference on Artificial Intelligence*, vol. 15, 1997, pp. 1340–1345.

[5] B. Yamauchi, A. Schultz, and W. Adams, "Mobile robot exploration and map-building with continuous localization," in *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, vol. 4. IEEE, 1998, pp. 3715–3720.

[6] I. Rekleitis, G. Dudek, and E. Milios, "Graph-based exploration using multiple robots," 2000.

[7] Y. Pei, M. Mutka, and N. Xi, "Coordinated multi-robot real-time exploration with connectivity and bandwidth awareness," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 5460–5465.

[8] C. Stachniss and W. Burgard, "Exploring unknown environments with mobile robots using coverage maps," in *International Joint Conference on Artificial Intelligence*, vol. 18, 2003, pp. 1127–1134.
[9] W. Burgard, M. Moors, C. Stachniss, and F. Schneider, "Coordinated multi-robot exploration," *Robotics, IEEE Transactions on*, vol. 21, no. 3, pp. 376–386, 2005.
[10] S. Sariel and T. Balch, "Efficient bids on task allocation for multi-robot exploration," in *The 19th International FLAIRS Conference*, 2006.
[11] R. Kitchin, "Cognitive maps: What are they and why study them?" *Journal of environmental psychology*, vol. 14, no. 1, pp. 1–19, 1994.
[12] B. Poucet, "Spatial cognitive maps in animals: New hypotheses on their structure and neural mechanisms," *Psychological Review*, vol. 100, no. 2, pp. 163–182, 1993.
[13] E. Tolman, "Cognitive maps in rats and men," *Psychological review*, vol. 55, no. 4, pp. 189–208, 1948.
[14] J. Gould, "The locale map of honey bees: Do insects have cognitive maps?" *Science*, vol. 232, no. 4752, pp. 861–863, 1986.
[15] B. Tversky, "Cognitive maps, cognitive collages, and spatial mental models," *Spatial Information Theory A Theoretical Basis for GIS*, pp. 14–24, 1993.
[16] S. Vasudevan, S. Gachter, V. Nguyen, and R. Siegwart, "Cognitive maps for mobile robots–an object based approach," *Robotics and Autonomous Systems*, vol. 55, no. 5, pp. 359–371, 2007.
[17] E. Topp, H. Huettenrauch, H. Christensen, and K. Eklundh, "Bringing together human and robotic environment representations-a pilot study," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*. IEEE, 2006, pp. 4946–4952.
[18] J. Cohen, S. McClure, and A. Yu, "Should i stay or should i go? how the human brain manages the trade-off between exploitation and exploration," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 362, no. 1481, pp. 933–942, 2007.