Cognitive Processing (2019) 20:203–225
https://doi.org/10.1007/s10339-018-0895-0

RESEARCH ARTICLE

What is a cognitive map? Unravelling its mystery using robots

Wai K. Yeap[1] · **Md. Hossain**[1]

Received: 17 January 2018 / Accepted: 3 December 2018 / Published online: 11 December 2018

Abstract
Despite years of research into cognitive mapping, the process remains controversial and little understood. A computational theory of cognitive mapping is needed, but developing it is difficult due to the lack of a clear interpretation of the empirical findings. For example, without knowing what a cognitive map is or how landmarks are defined, how does one develop a computational theory for it? We thus face the conundrum of trying to develop a theory without knowing what is computed. In this paper, we overcome the conundrum by abandoning the idea that the process begins by integrating successive views to form a global map of the environment experienced. Instead, we argue that cognitive mapping begins by remembering views as local maps and we empower a mobile robot with the process and study its behaviour as it acquires its "cognitive map". Our results show that what is computed initially could be described as a "route" map and from it, some form of a "survey map" can be computed. The latter, as it turns out, bears much of the characteristics of a cognitive map. Based on our findings, we discuss what a cognitive map is, how cognitive mapping evolves and why such a process also supports the perception of a stable world.

Keywords Spatial cognition · Cognitive mapping · View-based model · Computational model

Introduction

Since Tolman's (1948) seminal paper, there has been much interest in how spatial representations underlie sentient spatial cognition and in particular, the idea that human and some non-human species compute cognitive maps of their environment. There exists strong empirical evidence that what is computed is some form of a Euclidean map. For example, we have, in humans, from the sketch maps we produce to our abilities to orient to unseen places (e.g. Buchner and Jansen-Osmann 2008; Ishikawa and Montello 2006) and, in animals, from observing their foraging behaviour in the wild (e.g. Luhrs et al. 2009; Normand and Boesch 2009; Polansky et al. 2015) to their spatial behaviour in controlled experiments (e.g. Cohen and Bussey 2003; Gallistel 1990). The discovery of place cells, head cells and grid cells in the brain provides neurophysiological support that a Euclidean map is also computed (Jeffery and Burgess 2006; O'Keefe and Nadel 1978). Yet despite this evidence, the exact nature of such a map remains a mystery and this has led some researchers to argue against the idea that a cognitive map is computed (Benhamou 1996; Bennett 1996; Mackintosh 2002; Tversky 1993; Wang and Spelke 2002).

Part of the mystery is that a cognitive map is unlike a cartographic map whereby having computed it, one could read out where everything is located. The geographers and environmental psychologists cautioned against this notion of a cognitive map with an outcry in the 1970's, "a cognitive map is not a cartographic map" (Downs and Stea 1973). Their studies show, for example, that our perceived distance in the city is affected by seemingly unrelated factors such as presence of barriers, intersections in routes, direction of travel and time spent living in the environment (Crompton 2005; Montello 1997; Raghubir et al. 2011). Beck and Wood (1976), for instance, reported an experiment whereby they showed how tourists remembered two physically adjacent structures as being far apart because these structures were experienced as the last buildings in two different journeys starting from the same place but moving in opposite directions. Why is this the case if our cognitive map is a global metric map of the environment? If not, what kind of a map is a cognitive map?

✉ Wai K. Yeap
wai.yeap@aut.ac.nz

[1] Centre for Artificial Intelligence Research, Auckland University of Technology, Auckland, New Zealand

Sketch maps produced from investigating the acquisition of route knowledge in urban areas show that what is learned is not a metric map of the entire journey. Instead, these maps show local accuracy (some parts are accurate while others are either missing or inaccurate) and often such accuracy is found at the start and end of a journey (e.g. Golledge et al. 1985; Rovine and Weisman 1989). Subjects also learn a list of landmarks and places visited, the order in which they were visited and/or a sequence of actions taken at various choice points in the journey. In other words, what they learn is a fragmented metric map together with some form of a topological map (Downs and Stea 1973; Evans 1980; Lynch 1960). However, if a metric map is learned at the start of the journey, why isn't a full metric map of the journey learned? Why does one learn both a metric map and a topological map? Our map is also found to be systematically distorted due to various factors such as the regularization of turns and angles (see review: Koriat et al. 2000) and the alignment and hierarchical effects (see review: McDonald and Pellegrino 1993). While having a precise and detailed map would be over-empowering (i.e. we don't need to know exactly where everything is), is there a reason to distort what is learned?

The elusive nature of cognitive maps has led some to argue against the idea, and they offer alternative explanations to account for the behaviour observed. For example, for rats searching for food in a radial arm maze, Brown (1992) argues that they could have considered only those alleys not visited and thus would not need to use a spatial (metric) map. In the water maze problem, Benhamou (1996) argues that rats could use some orientation mechanisms and not a spatial map to locate the platform in the water maze. Bennett (1996) argues that many of the experiments that demonstrate the ability of finding novel shortcuts could be due to path integration rather than the use of a cognitive map. Even for humans, the idea has also been challenged. Wang and Spelke (2002) argue that our map comes from our symbolic reasoning about the environment and not from having evolved a different cognitive mapping process. More recent discussions that raised the possibility that insects have cognitive maps have also been debated along similar lines (e.g. Cheeseman et al. 2014; Cheung et al. 2014). Inferring the nature of the map from behavioural data is extremely difficult, especially when researchers do not have an adequate process model underlying cognitive mapping and cannot inspect the "program" in one's head.

To unravel this mystery, a computational model of cognitive mapping is needed to show what is computed and why. However, and unlike the modelling of perceptual processes, one faces a conundrum trying to develop a model without knowing what is computed in it. Note that the problem cannot be defined away given the controversial nature of the map. For example, while it has been observed that the map is inexact and incomplete, one does not know why and how inexact or incomplete the map is. Worse, some have argued that the map could be non-existent. Consequently, any attempt to define such a map and show how it is computed does not exactly resolve the mystery itself. To solve this conundrum, we seek a model that has explanatory power rather than demonstrative power (Yeap 2011). We note that empirical researchers have proposed and debated many hypotheses about cognitive mapping. For a model to be explanatory adequate, we need to demonstrate which of these hypotheses the model could support and why and in doing so, we highlight the inadequacy of some of these hypotheses, correct any misconceptions and provide further insights/predictions about cognitive mapping. In other words, unlike many existing models of cognitive mapping, our model is not developed specifically to account for some of these hypotheses. Rather, the model is developed independently and tested to see if these hypotheses could be verified using it. To achieve this, we empower a robot with the process and study its behaviour and just like studying the behaviour of different species, we observe its behaviour and discuss how cognitive mapping works.

We use the word "empower" to emphasize that our implementation of the model is not about algorithmic performance (i.e. how well or efficient it computes its map) or about achieving a naturalistic simulation (i.e. how similar it is when compared with a particular species). For the former, it means that what is computed in the model will be judged only loosely. For example, we create a process that computes an inexact and incomplete map, but we are not concerned with how inexact or incomplete the map is. Instead, we are concerned with how the system behaves when empowered with such a process. For the latter, it means that the system itself does not need to be biologically realistic. In our implementation, we use a wheel-based robot equipped with a laser and an odometer. It is unlike any other species but much of what is known about cognitive mapping has come from studying different species that have a variety of forms and use different sensors and methods to explore their environments. Our robot could be a species of its own, albeit an artificial one. If it exhibits cognitive-like behaviour and computes a cognitive-like map, we claim it is empowered with a process to compute its own cognitive map and, as we shall soon see, much could be learned from its process about cognitive mapping. We will refer to a robot designed for the study of cognitive processes without mimicking any particular species as an Albot (Yeap 2011).

In this paper, we present our study of cognitive mapping using the above approach. In "The process: a modular, view-based model of cognitive mapping" section, we develop a novel view-based model of cognitive mapping that does not compute directly a global metric map of the whole environment experienced. Instead, it takes views[1] as inputs whereby each view is used as a map of the local environment visited. A list of such views represents the path traversed. We show a global map of one's immediate surroundings could be obtained by integrating two or more adjacent views in the list. In "Albot$_1$: a robot empowered with a view-based process for exploring its environment" and "The experiments: analysing Albot$_1$'s map and process" sections, we empower a mobile robot with the process and conduct a series of experiments to investigate the nature of such global maps computed and the robot's behaviour as it acquires its own "cognitive map". In "General discussion" section, we show that our model is explanatory adequate by discussing what perplexing characteristics of cognitive mapping could be accounted for by our model and what further insights into cognitive mapping could be gained. In "Conclusions and future work" section, we conclude with a discussion of future work.

The process: a modular, view-based model of cognitive mapping

Currently there exist two process models for learning about the spatial environment. The first is based on a logical idea—a map is computed via the integration of information from successive views. The second is based on an empirical observation—many species are observed to remember snapshots (2D images) of a place and use them as part of their navigation strategy (Cartwright and Collett 1982; Graham and Collett 2002; Pecchia and Vollortigara 2010). Several algorithms have been proposed for the latter. They show how remembering snapshots helps insects to navigate towards their goal (e.g. Cheng 2008; Cheung et al. 2008), creates a topological network of arbitrarily defined but unique places (e.g. Kuipers and Byun 1991; Schölkopf and Mallot 1995) or a qualitative map of the environment (e.g. Wagner et al. 2004) and, more recently, enables route recapitulation (Gaffin and Brayfield 2016; Smith et al. 2007). Since snapshots cannot be used for orienting to unseen locations, this process is never referred to as cognitive mapping but rather as an alternative process for learning about the spatial environment and which is often proposed for insects.

The first process model, on the other hand, has been widely accepted by cognitive researchers as the model of cognitive mapping (e.g. Burgess 2006; Tatler and Land 2011; Wang and Spelke 2002). Cognitive researchers pay much attention to the frame of reference used when computing such a map and in particular, whether one computes an egocentric map or a non-egocentric map. It is generally accepted that an egocentric map is computed first, because it is a form that vision delivers and it is a form that is most natural for guiding action (Evans 1982). As the map gets larger, maintaining it would be daunting and at some point, the information would be transferred into an enduring non-egocentric map (Burgess 2006; Sholl 2001). Unfortunately, these researchers overlook two important aspects of this process that concern its use as a model of cognitive mapping. First, they fail to consider the serious effect of sensor noise that will render any such enduring map useless (for a more detailed discussion, see Yeap 2014). Interestingly, robotics researchers face a similar problem when implementing the process on robots and they identify correctly that the key problem to solve is simultaneous localization and mapping or "SLAM". They developed a neat probabilistic solution for SLAM that enables robots to compute an exact map of their environment (for a review of such works, see: Bailey and Durrant-Whyte 2006; Durrant-Whyte and Bailey 2006). Second, if, like the robotics researchers, one could overcome the distortions and compute a useful map, that map would be, in Wang and Spelke's (2002) words, "too all-embracing in scope and flexibility" and would be unlike a cognitive map. However, unlike Wang and Spelke who then argue against the idea that a cognitive map is computed, we argue that it is the process model underlying cognitive mapping that needs to be replaced.

McNamara and his co-workers (McNamara 2003; Mou et al. 2004; Rump and McNamara 2007) offer an interesting alternative model whereby views are remembered on their own as descriptions of places visited, using whatever intrinsic frame of reference (including but not necessarily limited to one's egocentric view) that is most suitable at the time of coding. These views are not updated with information from subsequent views of the same place unless the new view offers a better description, such as being less obstructed or aligned with more salient axes in the environment. If this happens, the view is then replaced by the new view. Unfortunately, they offer the same mechanism of updating continuously one's position and orientation in these views so that a global metric map of the environment traversed can be computed. As noted above, without correcting sensor noise, such a map would be too distorted or, with sensor noise corrected, too exact and detailed as a cognitive map.

Our model

In developing our model, we observe that a view shows not just what things are in it but also where things are in its bounded space. Taking a view as input, one instantly has a map of the local environment that one is about to explore and could use it to guide one moving in it. For example, one could use it to decide where to retreat to if a predator

[1] Each view is delivered by its sensor(s), which is not necessarily vision-based and it describes what and where things are out there.

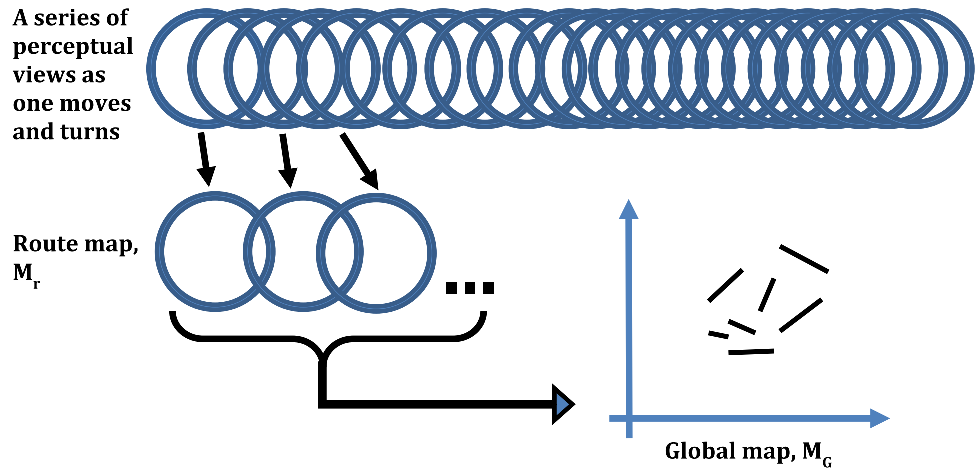

Fig. 1 A basic cognitive mapping process that computes first a route map, M_r, which is a series of perceptual views each denoting a local environment visited. From it, some form of a global map, M_G, can then be generated

suddenly appears or where to go next. By tracking local objects in subsequent views, one could triangulate one's position in the map using them, thereby localizing oneself in the map. By assuming that the world one lives in is stable, the input view/local map does not need to be updated as one moves in it. This is because nothing significant would have changed and new objects appearing in subsequent views are ignored as they are deemed to lie outside the map. When no local objects are in view, one has exited the current local environment. One then takes another view as input and uses it as a map for the next local environment that one is about to enter. This process repeats itself as one moves about in the environment.

Formally, our model is defined as follows: (1) the process takes the current view as a map of the local environment that one is about to explore; (2) while moving in it, the process tracks local objects in view and (3) when no local objects are in view, repeat step (1). Our model thus computes directly a list of views (henceforth also referred to as local maps) rather than a single global metric map. Just like the McNamara et al. model, views remembered are not updated as one moves in their bounded space. However, unlike in the McNamara et al. model, none of the views is to be replaced by a "better" subsequent view and there is no continuous tracking of one's position in it. For species that do not need to return "home", earlier maps can be "forgotten" as one continues to explore one's environment. If not, a collection of such views represents the path one took through the environment and could be used as input for computing a richer description of the environment. Computing it first as part of cognitive mapping, however, is reminiscent of an old dictum in cognitive mapping: "a route map is computed first, prior to a survey map (i.e. a global map)". Henceforth, we will refer to such a list of views as a route map.

Being view-based, our model is attractive as a model of cognitive mapping but in addition, our model is also modular in Fodor's (1983) sense, (see also Barrett and Kurzban 2006; Mandelbaum 2013). Computing a route map is almost free: one's perceptual system delivers the views required automatically and already has various mechanisms for tracking objects across views, say, for the perception of persisting objects (Scholl 2007). The process is thus fast and domain specific and as noted above, repeats itself continuously. The process is also automatic and ballistic and, as Mandelbaum (2015) argues, this means that the process is information encapsulating.

Could computing a route map provide a sufficient basis for learning much of the characterizations of cognitive mapping found in the literature? A central thesis of cognitive mapping is that one also learns some form of a fragmented global metric map that is inexact and incomplete and that allows one to orient oneself in the environment. As noted earlier, one way to do so is compute a route map with overlapping views (see Fig. 1). In the next two sections, we empower $Albot_1$ with such a process and conduct three sets of experiments to study its mapping behaviour as it explores its environment.

$Albot_1$: a robot empowered with a view-based process for exploring its environment

In this section, we present three key algorithms for empowering our robot, $Albot_1$, with a view-based process of its own (Fig. 1). $Albot_1$ is a pioneer-3DX mobile robot (from MobileRobots Inc., Amherst, NH, USA), equipped with a 180-degree SICK laser scanner and an odometer. $Albot_1$ computes simultaneously a route map and a global map of its environment as it explores in an office environment. The three key algorithms required are: (1) an algorithm for tracking surfaces across views, (2) an algorithm for adding local maps to its route map and (3) an algorithm for adding and deleting information from its global map.

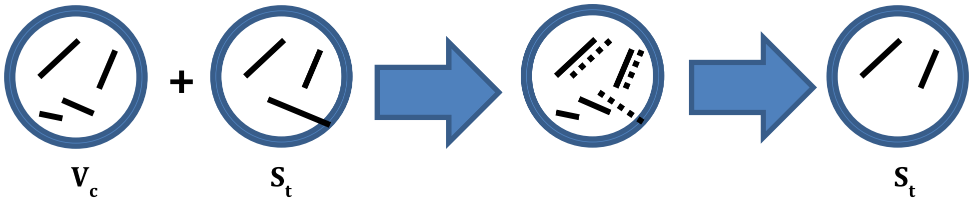

Fig. 2 Albot$_1$'s algorithm for tracking surfaces: the surfaces to be tracked, held in S_t, are transformed (+) onto the incoming view, V_c, using the distance moved and the angle turned to get to V_c. Any tracked surface (dotted lines) found "close" to a surface in V_c is identified as the same surface. These then become the new set of surfaces to be tracked and described using the coordinate of V_c

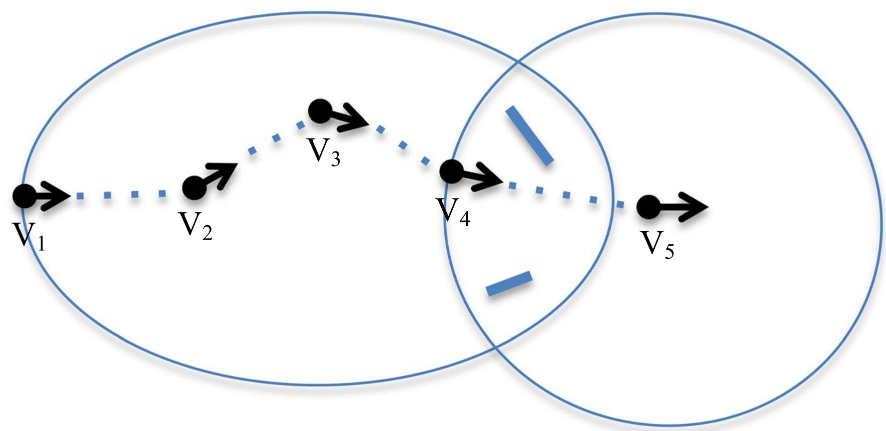

Fig. 3 Creating a route map and a global map: $V_1 \ldots V_5$ are five robot viewing positions and the ellipses indicate the view boundary for V_1 and V_4. A route map created using V_1 and V_4 (as opposed to V_1 and V_5) would allow one to triangulate the viewing position of V_4 in V_1 using the tracked surfaces (solid line) perceived in view V_1, thereby allowing the creation of a global map that combines views V_1 and V_4

Tracking local surfaces

The inputs for Albot$_1$ are laser points that are converted into lines denoting surfaces in view. To compute a route map, Albot$_1$ tracks surfaces belonging to its current local environment that are still in view. However, since Albot$_1$ only uses these surfaces for triangulating its position in its local environment (see "Computing simultaneously a route map and a global map" section), we implement an algorithm that tracks only those surfaces that are deemed best for the triangulation task. Such surfaces would exclude tiny surfaces, surfaces that can be distorted easily due to perspectival viewing and surfaces without any occluding point. Albot$_1$ thus uses the following three criteria to select surfaces for tracking: (1) it is of a reasonable length (> 40 cm), (2) it does not lie parallel to the robot's facing direction (i.e. the robot's viewing angle of the two endpoints of the surface is no less than 3°) and (3) it has at least one endpoint that has not been occluded. All threshold values are selected arbitrarily.

Since Albot$_1$ perceives 2D surfaces as lines, there are not many useful features that one can use for the recognition task. Consequently, we implement the standard algorithm of transforming information in one view to the next and finding co-locating surfaces as a method of "recognizing" surfaces. A surface is "recognized" as the same if the distance between the endpoints of two co-located surfaces is less than 40 cm and their view angles are no more than 5 degrees.

Figure 2 outlines how the algorithm works (for its pseudocode, please see "Appendix 1"). Using this algorithm, some of the tracked surfaces could be missed but what this means is that Albot$_1$ will lose track of these surfaces earlier than expected. However, this does not affect the process in any significant manner.

Computing simultaneously a route map and a global map

We observe that a series of local maps (i.e. views remembered) can be combined to form a global map if the positions of the origin of each local map in its previous local map are known. This condition can be achieved if every local map is created prior to exiting its current local map and at a point where some of the tracked surfaces are still visible. The latter enables one to triangulate one's position in each local map (Fig. 3). Albot$_1$'s route map is computed as such and is represented as a list of tuples (V_i, $R^i_{x,y}$), where V_i is the view remembered and $R^i_{x,y}$ is the coordinates of the robot's viewing position in the previous local map, V_{i-1}.

Albot$_1$'s algorithm for computing both its route map and global map can now be described. Using the tracking algorithm (Fig. 2), Albot$_1$ continuously tracks surfaces as it moves in its local environment. Whenever Albot$_1$ has less than four tracked surfaces, S_{ti}, in its current view, V_c, it adds a new local map, (V_c, $R^c_{x,y}$), to its route map (Fig. 4). The

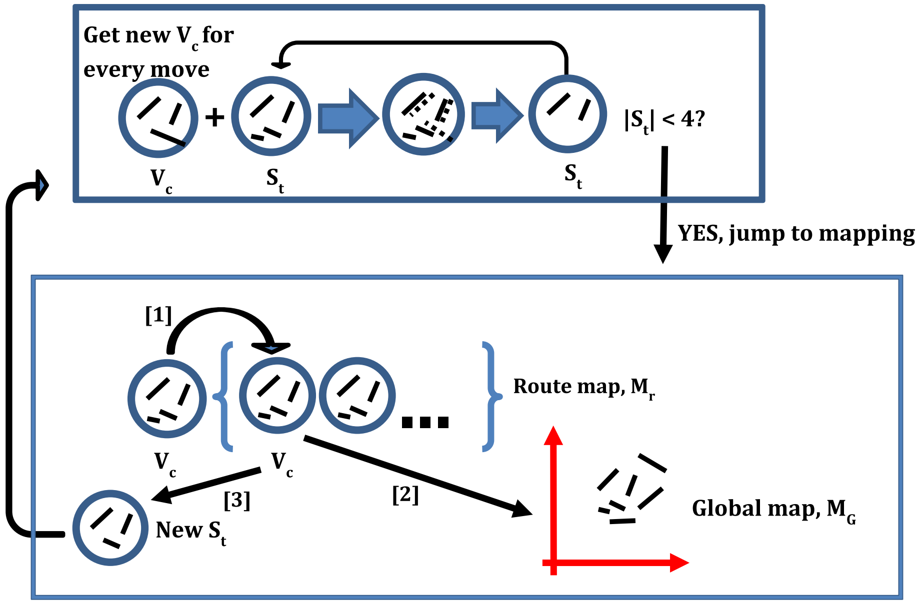

Fig. 4 Albot$_1$'s main routine (basic idea): The top box is a perceptual process that delivers a new view, V_c, every time one moves and turns and then tracking surfaces, S_t, in it. The bottom box is the mapping process whereby a new view, V_c, is added to the route map (step [1]) and to the global map (step [2]). A new set of surfaces to be tracked is then computed (step [3])

coordinates of the surfaces in V_c are then transformed into that of the global map and these surfaces become part of the global map. However, we found that it is possible that one could end up with no trackable surfaces in the current view and as such Albot$_1$ would not be able to localize itself in the current local map prior to entering the new one. A straightforward solution would be to always use the previous view (V_{c-1}, $R^{c-1}_{x,y}$) instead, i.e. it remembers the previous view as the next local map since it guarantees that one could localize oneself in it. In the example given in Fig. 3, the route map will consist of the views V_1 and V_3 together with their respective coordinate positions and not V_1 and V_4.

Unfortunately, in the above situation, one finds oneself immediately in a local environment with few (<4) or no trackable surfaces in view. This means that Albot$_1$ would need to immediately update its map again, but it could not now use V_{c-1}; otherwise, it would be in a loop. A switch (a Boolean variable, used-V_c,—see "Appendix 2" for details) is used to signal that one should use V_c instead of V_{c-1}. Note that in the situation where V_c has no trackable surfaces, Albot$_1$ would not be able to localize itself in the map using triangulation and a simple alternative would be to use path integration. That both mechanisms, path integration and triangulation, are needed in Albot$_1$'s process is interesting from a cognitive mapping standpoint since both strategies are widely used among species (e.g. Zhao and Warren 2015). Figure 5 shows Albot$_1$'s updated mapping process that includes the use of path integration.

"Appendix 2" provides the pseudo-codes for these three algorithms: the main algorithm, an algorithm for updating the maps and an algorithm that updates the maps using path integration. Briefly, to localize the origin of each local map (i.e. one's current position) in the previous local map, Albot$_1$ thus uses either path integration or triangulation. The former method is a straightforward method using vector addition and the latter method requires selecting a suitable (or the best) trackable surfaces, S_{t1},…S_{tn}, that are in view for triangulation. To do so, Albot$_1$ uses two heuristics: one for ruling out which S_{ti} cannot be used and one for choosing the best remaining S_{ti} to triangulate its position. The first heuristic involves computing the average localization point and its standard deviation from all points afforded by each tracked surface. Any S_{ti} that generates a localization point whose distance from the average localization point is greater than the standard deviation is then ignored. The rationale is that these points, if not distorted, should be located close together. The second heuristic for deciding which of the remaining points best represents Albot$_1$'s position is as follow: observe that Albot$_1$ explores its environment using linear motion, always knowing how much it turns and moves and therefore it can roughly localize itself in the current local environment using path integration. One simple heuristic then is to choose the

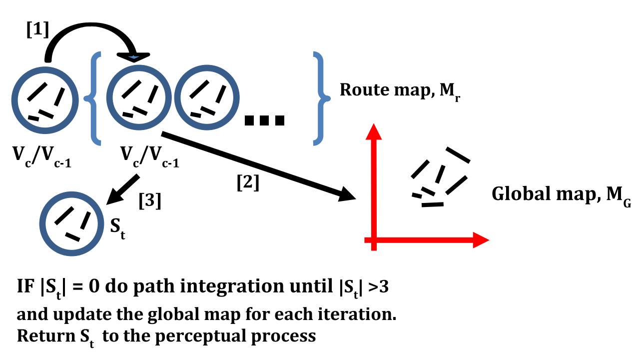

Fig. 5 Albot$_1$'s cognitive mapping process

point generated by the S_{ti} closest to Albot$_1$'s position computed via path integration. Once Albot$_1$'s current position in the previous local map is computed, the current view can now be added to the route map and its content transferred to its global map.

Updating the global map

When updating the global map, an immediate problem arises: how should overlapping information be dealt with? For robot mapping, such information is often combined using a best-fit strategy to produce a complete and exact map. For cognitive mapping, doing so would enable one to know that one is returning to a familiar place. However, without correcting sensor errors, overlapping parts do not necessarily refer to the same part of the physical environment and as such, any attempt to combine them is futile. By following the principle of no updating of local maps, we use an alternative and a more parsimonious strategy whereby overlapping information is simply deleted rather than merged. Albot$_1$'s algorithm is described below (for its pseudo-code, see "Appendix 3").

First, all surfaces in the global map that fall within a slightly enlarged spatial extent of the incoming map are identified. We use an area slightly larger than the current view to remove a bit more of the surfaces in front. The rationale is that these surfaces, if any, are unlikely to be a useful part of the current memory. Surfaces are removed using the following rules: if they belong to the current map, they are removed immediately (this is the case of overlapping surfaces between the current map and the incoming map as a result of moving forward); if they belong to past local maps, either the whole or parts of these old maps are removed. For example, when surfaces belonging to local maps 25[2] and 28 are found, we remove all surfaces belonging to local maps from 25 to 28 that fall on the facing side of the robot. The rationale for doing so is that these maps are not experienced in isolation and thus should be removed as a group. However, only those surfaces belonging to these maps that Albot$_1$ could "see" from its current position are removed. Doing so leaves more information in the global map, thus creating a residual effect. Without combining incoming views with information in the global map, it turns out that our approach is limited in that Albot$_1$ does not know that it is returning to the same part of the environment; it keeps deleting information in the global map whenever it "moves" into a part of it thereby creating a transient global

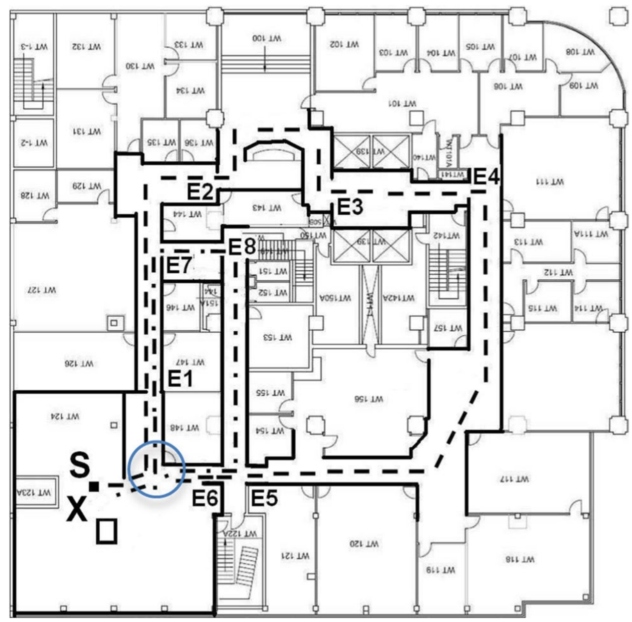

Fig. 6 A typical test environment

map. However, we argue in the discussion section that recognition of places visited earlier is best performed using a separate process.

The experiments: analysing Albot$_1$'s map and process

Three sets of experiments were conducted using Albot$_1$ to investigate the cognitive aspects of both our process and the global map computed. In particular, we ask the following questions: (1) can Albot$_1$ compute a useful global metric map?, (2) if so, what kind of a global map is computed? and (3) how does Albot$_1$'s process compare with the traditional process of integrating successive views to form a map? To provide a more insightful analysis, we conduct our experiments together with a SLAM-based robot. While the SLAM implementation is never intended to be a model of cognitive mapping, it is nonetheless the only successful realization of a mapping process favoured by cognitive researchers and as such, a comparison with Albot$_1$'s process can provide an interesting contrast between the two different approaches from a cognitive mapping standpoint. The SLAM method we have chosen is Carmen mapping (Thrun et al. 2001), sourced from: http://carmen.sourceforge.net/) but, as noted, the particular choice of algorithm is unimportant in our study. Any shortcomings identified here concern only the process itself as a candidate model for cognitive mapping and not as an algorithm for robot mapping. To de-emphasize the particular algorithm used, we will refer to the robot itself, as Albot$_x$, and not to the algorithm.

[2] Surfaces belonging to the same local map are tagged with a unique identity.

Fig. 7 Maps computed by **a** Albot_x and **b** Albot_1 in Experiment 1

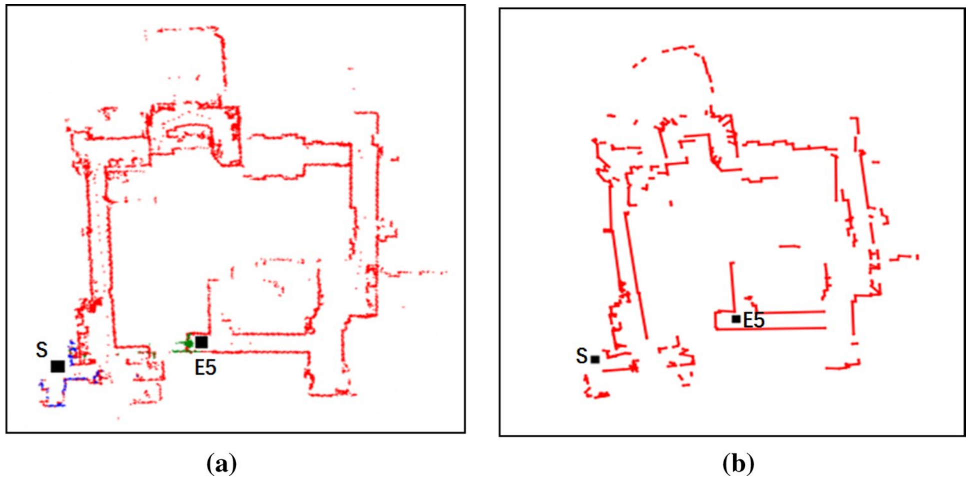

Figure 6 shows the main test environment used and a path, $S \to E1 \to E2 \to E3 \to \cdots \to E6 \to E1 \to E7 \to E8 \to E6 \to X$, taken by the robot. S and X refer to the same physical location. The robot is either driven continuously throughout the environment along the chosen path or moved in a start–stop manner. Note that there are other objects in the environment (e.g. chairs and tables) and therefore the layout depicted in Fig. 6 does not represent the exact environment that the robot has experienced. Using this particular environment, we collected three datasets by driving the robot continuously using the path shown. Dataset 1 consists of 1330 views, where consecutive views are recorded either at 10 cm or 1 degree apart. Dataset 2 is the same as dataset 1 except that at each step the odometer data have an extra 10 cm or 1 degree added to induce errors. Dataset 3 consists of 217 views, where consecutive views are recorded either 1 m or 10 degree apart. Two other environments will also be used and will be introduced later.

Experiment 1: can Albot_1 compute a useful global metric map?

Many species can orient to nearby places and what cognitive mapping tells us is that they use a map to do so. It is important to show in our first experiment that the global map computed by Albot_1 is good enough for it to orient itself in its environment despite no correction of sensor errors. Given that different species rarely orient exactly to where they want to go (for example, see Etienne et al. 1998), it suffices that Albot_1 produces a "good enough" map that shows only a rough shape of the environment traversed.

Figure 7 shows the maps computed by Albot_1 and Albot_x for the path $S \to E1 \to E2 \to E3 \to E4 \to E5$. The robots traversed approximately 100 metres (measured by their odometers) and obtained 795 views as recorded in dataset 1. From the 795 views, Albot_1 updated its global map only 307 times (i.e. it remembers 307 views), whereas Albot_x updated its map 795 times. While Albot_x produced a slightly inexact map, it will be able to correct itself when it continues to move towards S, thereby creating an exact map. Albot_1's map, on the other hand, is inexact (i.e. it will not be able to correct itself) and incomplete (parts of the environment are missing from its map). Nevertheless, by inspection, Albot_1 is able to use its map to orient to some nearby places.

However, just to demonstrate that this is the case, we measure the distance and orientation of its starting position from two different stop positions in two separate journeys (Fig. 8). In this test, Albot_1 explored its environment using a start–stop approach so that its exact stop positions can be selected. Corresponding measurements were then calculated using the floor plan of the physical environment (see Fig. 6). The results are shown in Table 1. Note that the orientation angle is the angle between Albot_1's current facing position and the starting point; its sign tells whether the angle is measured with Albot_1 turning left (+) or right (−). Given that Albot_1's measurements deviate no more than $\pm 7°$ and ± 3 m, the results confirm that Albot_1 performs very well in this test. However, as we shall soon see, Albot_1's map is not a complete and precise map and as the map grows in size, some parts can no longer be oriented properly. In contrast, Albot_x produces an exact map for the entire environment traversed and hence will be able to orient precisely to any parts of its map.

We also tested Albot_1 using a dataset downloaded from the robotics data set repository (Howard and Roy 2003). This extra dataset is chosen because the robot does not explore its environment by moving forward continuously; it turns back frequently and returns to places visited earlier. This tests if Albot_1's global map would become too distorted as it continuously adds and removes local maps from its vicinity. Figure 9a shows the map produced by Albot_x using this dataset (but with a different SLAM algorithm, GMapping from Grisetti et al. (2007)). The robot's path is $S \to 1 \to 2 \to \ldots \to 17 \to E$. Figure 9b shows that the map

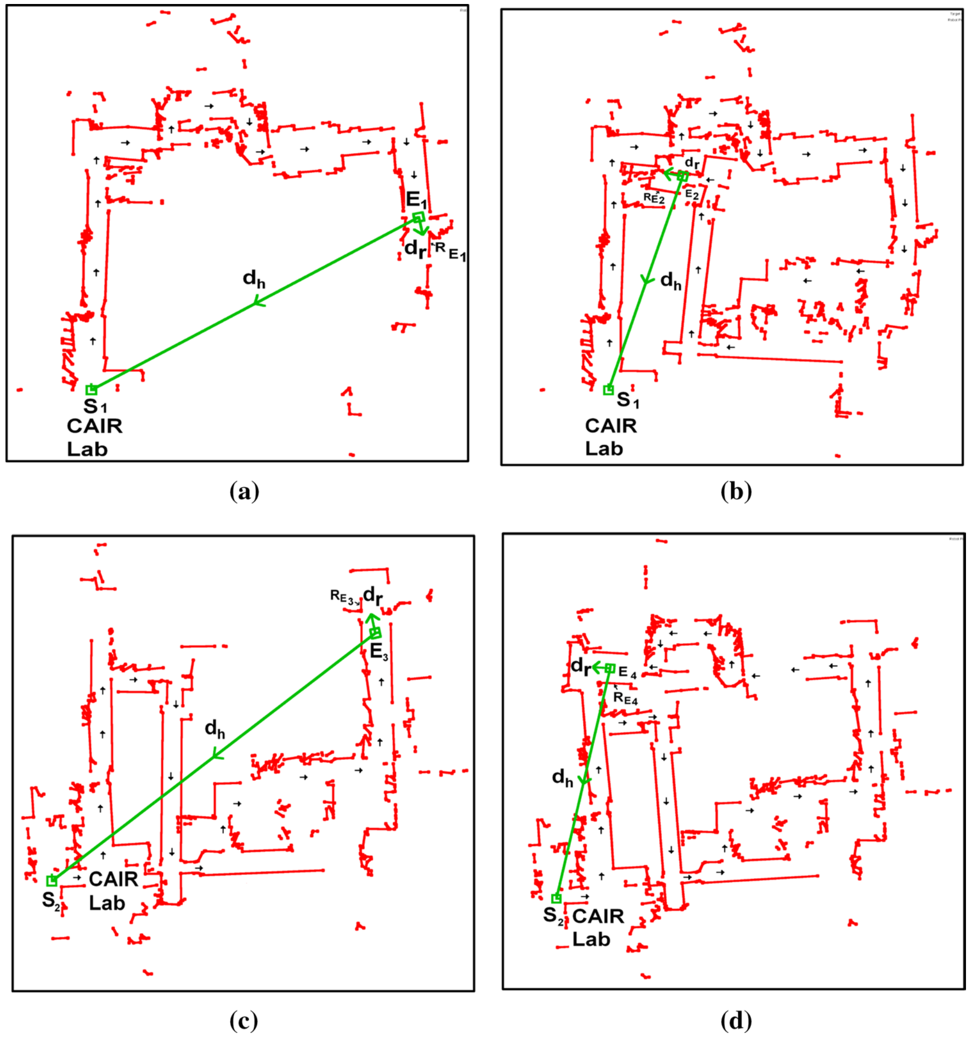

Fig. 8 Calculating the distance and orientation in two different journeys: (top) Albot$_1$ explores the environment (Fig. 6) in a clockwise direction and (bottom) in an anti-clockwise direction

Table 1 Asvdv

	Albot$_1$'s map	Physical map
Figure 4a	−75.5°/25.8 m	−70.1°/27.3 m
Figure 4b	83.0°/19.6 m	77.0°/17.0 m
Figure 4c	114.0°/32.1 m	117.5°/35.3 m
Figure 4d	82.5°/21.6 m	76.0°/23.0 m

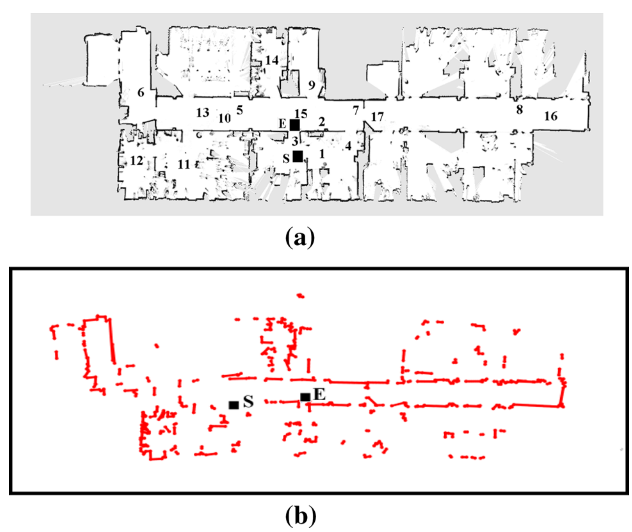

Fig. 9 Maps computed by **a** Albot$_x$ and **b** Albot$_1$ using a dataset from a repository

produced by Albot$_1$ still could represent the approximate shape of the environment traversed. However, in absolute terms, Albot$_1$'s map is seriously distorted since the point E is not where it should be (with respect to S which is plotted in its original position). However, what this shows is that in Albot$_1$'s map what matters most is not the absolute position of where things are but whether its overall shape corresponds well to the overall shape of the environment traversed. The latter would allow it to orient itself in its immediate surroundings.

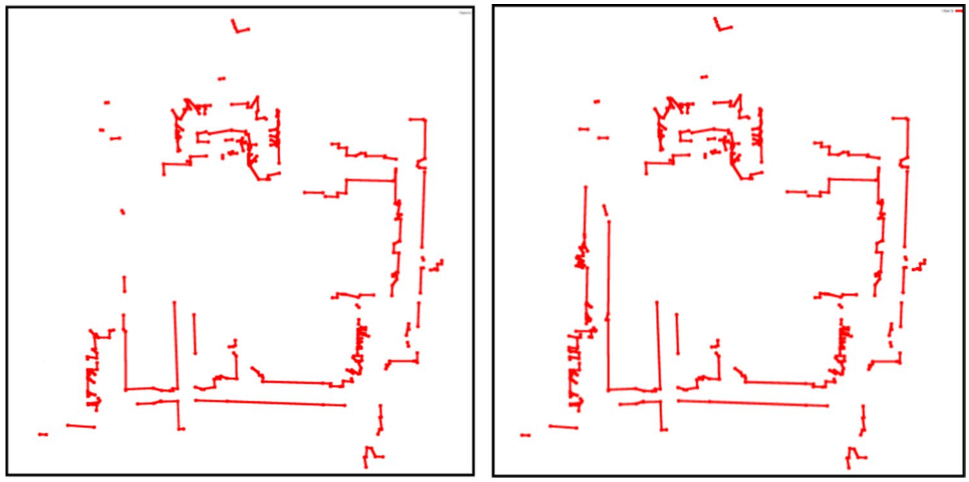

Fig. 10 Maps of Albot$_1$ produced as it re-visits parts of the environment as shown in Fig. 6

Experiment 2: what kind of a global map does Albot$_1$ compute?

Albot$_1$'s global map is dynamic due to the deletions of information whenever incoming information overlaps with existing information. The underlying cause for the latter could be due to a familiar part of the environment being re-visited or, simply, the overlapping of unrelated parts of the map due to spatial distortion. To highlight the map's dynamic nature, we continue with Albot$_1$'s exploration of its environment from where it stops in Fig. 7b. As Albot$_1$ moves past the point marked with a circle in Fig. 6, much of the corridor on the left that was in the map earlier (Fig. 7b) has now been removed (Fig. 10a). As Albot$_1$ continues, it re-builds the corridor from a fresh perspective (Fig. 10b). The map computed by Albot$_x$ would remain the same (map not shown).

Instead of computing a map of the entire environment experienced, Albot$_1$ computes a map that shows only how local maps experienced earlier are related to its current local environment. Note that local maps experienced further back in the route may or may not be displayed and if displayed, could be seriously distorted. For example, observe how the map changes from Figs. 7b, 8, 9 and 10a. Figure 7b shows that the points S and E are seriously misaligned but are not so in Fig. 10a, when Albot$_1$ has re-visited the point S after having traversed past the point *E*. Albot$_1$'s global map is thus path-dependent and "egocentric". Note that the term egocentric is not used here in its mathematical sense that many researchers, both in the neural and cognitive sciences, have used to describe a cognitive map (e.g. Ekstrom et al. 2014; Meilinger and Vosgerau 2010). Instead, it is used to describe a map whose information can only be interpreted from the current local map that one is in as opposed to one whose information could be referenced independently from any position in the map. The latter would be described as an allocentric map and an example would be Albot$_x$'s global map.

So far, we have shown Albot$_1$ is capable of generating a single useful map for the entire environment explored. However, as noted above, this should not always be the case as sensor errors can cause older parts of the map to be seriously misaligned. This is demonstrated in the next experiment in which Albot$_1$ explores a route that is not corridor-like (Fig. 11a). Here the robot is driven continuously from *S* to *E* via X over a distance of approximately 130 metres. It collected 153 views in this journey and updated its global map 103 times. The result shows that at point *X*, Albot$_1$ maintains a reasonable map of the environment traversed (Fig. 11b) but at point E, the earlier parts of its map have become seriously distorted (Fig. 11c). Thus, what Albot$_1$ could learn is not a single global map but a series of independent but connected global maps. For example, in this test, it could learn two global maps; one at point *X* (Fig. 12a which is the same as Fig. 11b) and another at point *E*, where what was computed up to the point *X* were removed (Fig. 12b). In contrast, Albot$_x$ learns a complete single global map of the environment (Fig. 12c).

Experiment 3: how does Albot$_1$'s process differ from Albot$_x$'s?

Using our terminology above, a key difference between the two approaches is that Albot$_x$ combines successive views to form an allocentric map (i.e. a global map of its environment experienced), whereas Albot$_1$ combines successive local maps to form an egocentric map (i.e. a global map of a segment of the path traversed starting from its current local environment). For the former, what robotics researchers have discovered is that if the robot maintains an estimate of the joint probability of its map and its poses and updates them with every successive view, these estimates can converge

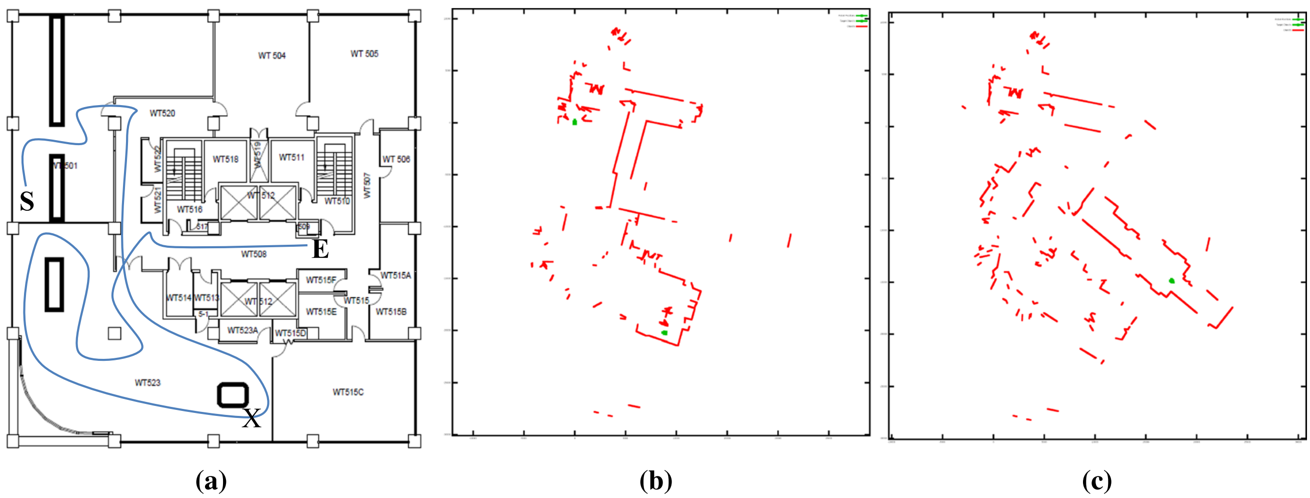

Fig. 11 Albot$_1$'s maps at points **b** X and **c** E when traversing (**a**) the environment

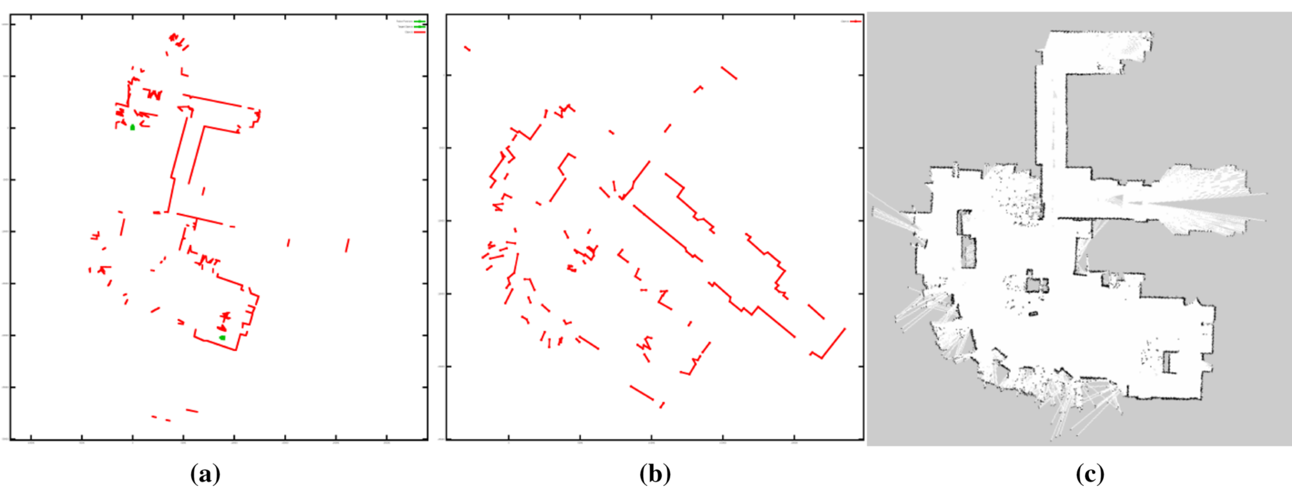

Fig. 12 Global maps learned: **a** and **b** Albot$_1$ learns two independent but connected global maps and **c** Albot$_x$ learns a single complete global map

and produce an accurate map, provided it could also recognize returning to places visited earlier. Such a process is thus computationally intensive, as it requires correct matching of landmark features and continuous updating of both the map (with incoming information) and the estimates of the positions of all the landmark features and the agent's poses in it. Albot$_1$'s process, on the other hand, is a much more parsimonious process, requiring many fewer updates and less attention to incoming views.

To demonstrate, we conduct an experiment using dataset 3 whereby each robot traverses the entire path, $S \rightarrow E1 \rightarrow E2 \rightarrow E3 \rightarrow \cdots \rightarrow E6 \rightarrow E1 \rightarrow E7 \rightarrow E8 \rightarrow E6 \rightarrow X$, for approximately 136 metres but is only given 217 views. In other words, the incoming views are not being processed as frequently as before or, more precisely, each consecutive view is either 1 m or at least 10 degree apart. Figure 13a shows Albot$_1$ still produces a reasonable map with the reduced input while Albot$_x$'s map is now seriously distorted (Fig. 13b). From the 217 views, Albot$_1$ updated its global map only 133 times. Albot$_x$'s process is thus not only computationally intensive but also rigid. Without continuous updating, it fails to "close the loop" and produce a correct map.

Albot$_1$'s map can be affected by what is in the environment. For example, if more trackable surfaces are perceived at a distance, Albot$_1$ will update its map less often and vice versa. Or, when no trackable surfaces are in view, Albot$_1$ shifts to using path integration rather than triangulation for localizing itself, thereby causing more distortions to the map computed. To demonstrate these effects, we analyse some

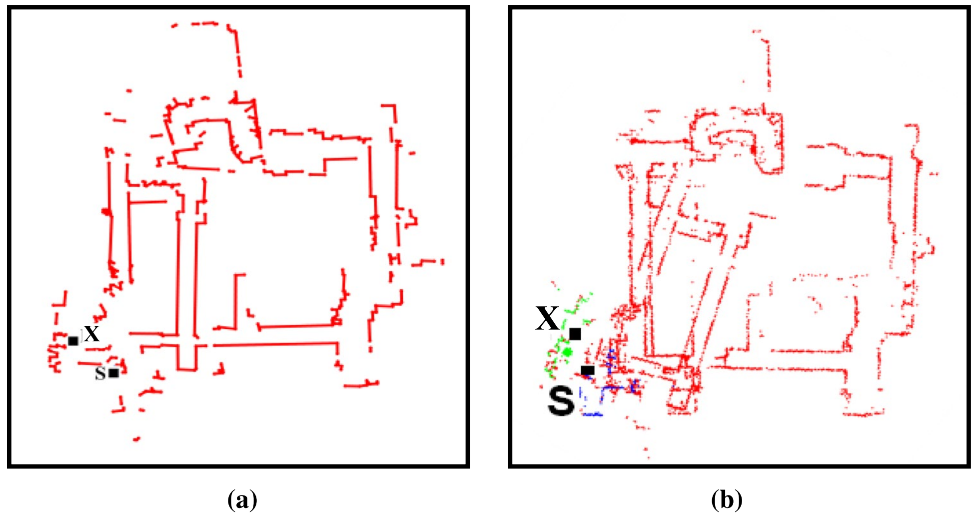

Fig. 13 Maps computed by **a** Albot$_1$ and **b** Albot$_x$ in Experiment 3

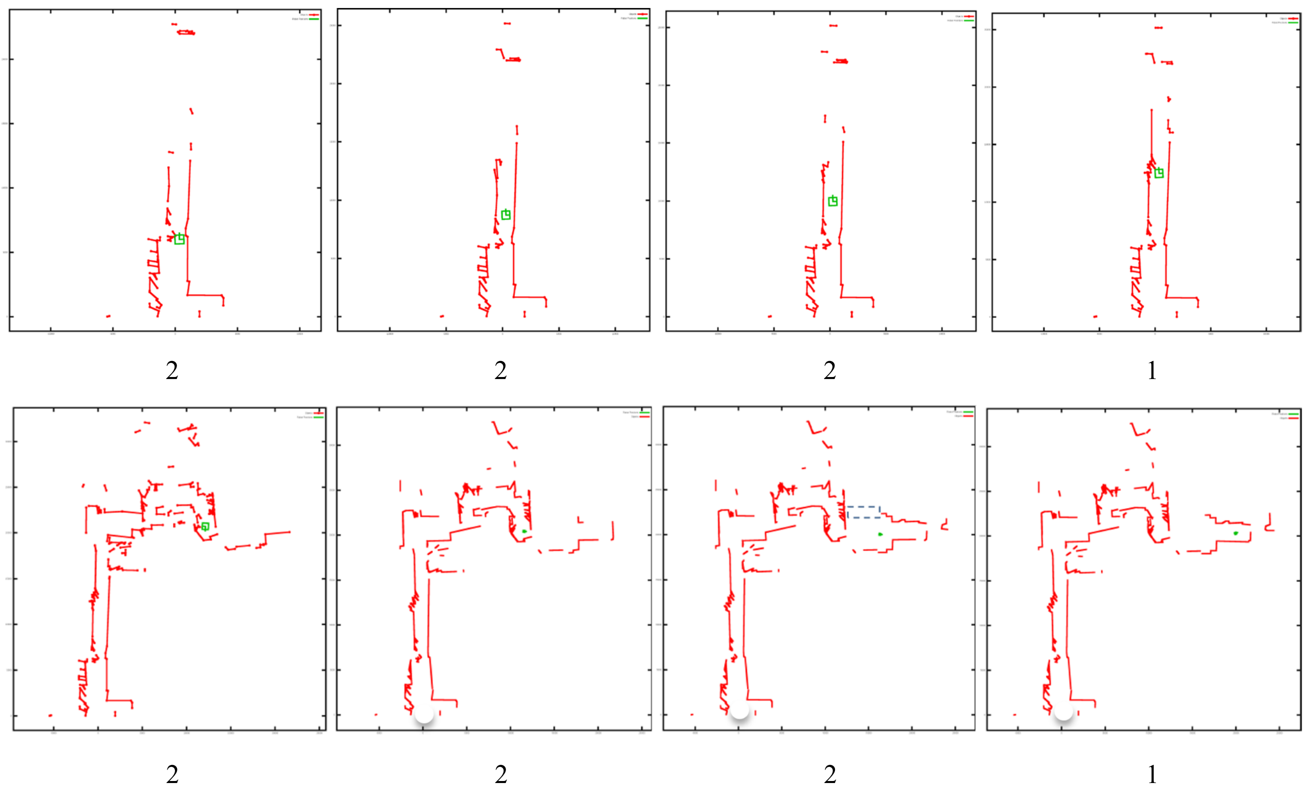

Fig. 14 Two different traces showing how Albot$_1$'s global map (top and bottom) is updated. The green marker shows the robot's position when updating takes place. Each number below indicates the number of move/turn instructions executed before an update is required. Each move is on average about 3 m

traces of Albot$_1$'s process. Figure 14 shows two traces of how Albot$_1$ updates its global map using the triangulation method. Albot$_1$ is moving through the environment as shown in Fig. 6 using a start–stop approach. The first trace (top row) shows how the map is updated when Albot$_1$ is moving through a corridor (left part of Fig. 6) and the second trace (bottom row), when Albot$_1$ is moving through a more open spaced area (top part of Fig. 6). In the first trace, trackable surfaces were seen close to Albot$_1$ and Albot$_1$ thus updated its map frequently (after at most 2 moves). This is because Albot$_1$ quickly lost track of these surfaces. However, the map generated is detailed, i.e. nothing much is left out. In the second trace, Albot$_1$ updated less often (after at least 2 moves) and the map is less detailed. For example, in the

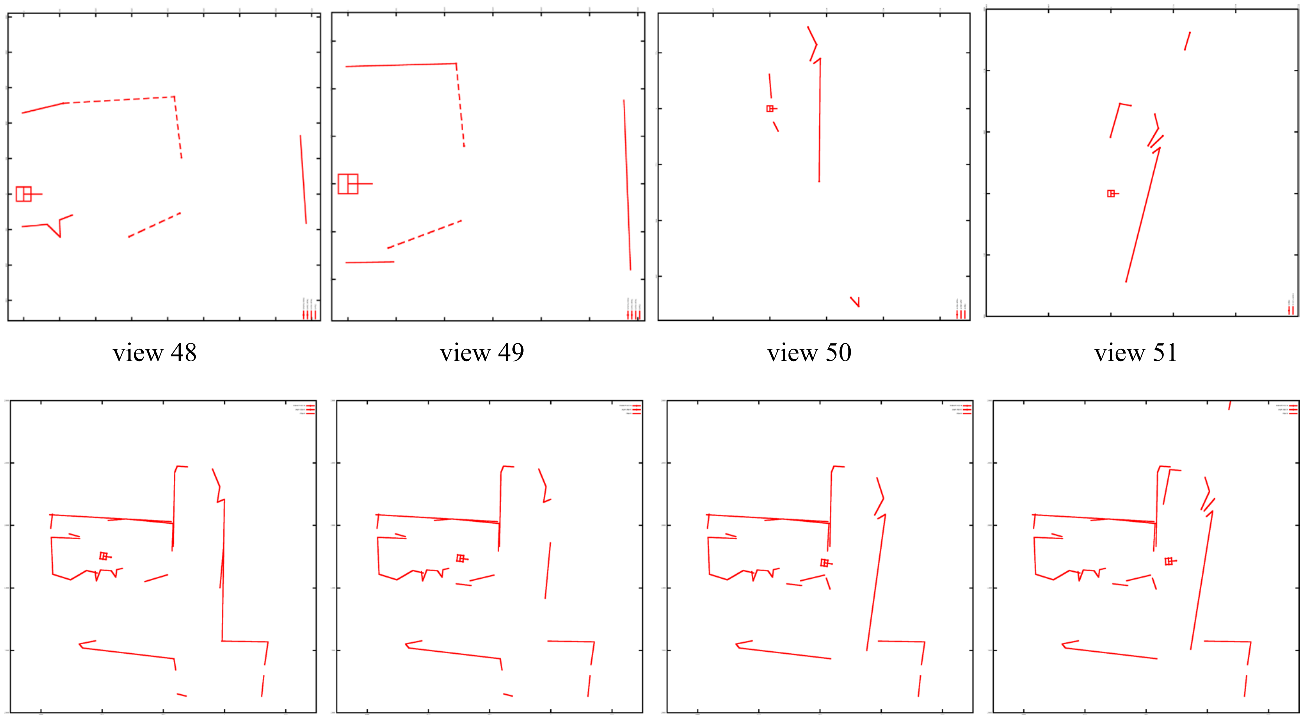

Fig. 15 Updating one's global map using both triangulation and path integration: (top) views perceived at 4 successive steps and (bottom) Albot_1's global map after updating at each corresponding step

third update moving through that part of the environment, a large area (marked by the dashed rectangle) is now missing despite being perceived by Albot_1.

Figure 15 shows Albot_1 reversing out of a room in the environment as shown in Fig. 6. Here, it updates its global map using triangulation and path integration. In view 49, Albot_1 sees only two tracked surfaces (dotted line) in view. This triggers it to update its global map using view 48 but since view 48 has only 3 tracked surfaces in view, it has already been used to update the global map. Hence, Albot_1 updates its map using view 49 instead and using triangulation (see bottom row). In view 50, no trackable surfaces are in view and Albot_1 thus continues to update its global map using view 50, but this time, it uses path integration. In view 51, Albot_1 continues to use path integration to update its global map using view 51. Note that using path integration, the map is updated after every move and consequently, errors could be added to the global map at a faster rate (compare the global map on the far left with the one on the far right in Fig. 15 and one sees that the right-hand wall now becomes more slanted).

In our final experiment, we look at the effects of sensor errors on path integration. We compare Albot_1's process with Albot_x's process that now integrates successive views to form a map without correcting sensor errors. In this experiment, we use two datasets, dataset 1 (without induced errors) and dataset 2 (with induced errors). The path taken by the robot is $S \rightarrow E1 \rightarrow E2 \rightarrow E3 \rightarrow E4 \rightarrow E5 \rightarrow E6 \rightarrow E7$. Figure 16 shows the results of this experiment. Without correcting sensor errors, Albot_x's maps are now seriously distorted (Fig. 16a, b). Albot_1's maps, however, still maintain the shape of the environment traversed (Fig. 16c, d). Note that in Fig. 16d, Albot_1's map is slightly more distorted than in Fig. 16c and this is due to the effect of using path integrations more often to update its map (7 times as opposed to 5 for Fig. 16c).

Final remarks

Despite no error corrections and no continuous integration of successive views, Albot_1 is able to compute a global metric map that allows it to orient to nearby places. However, unlike a cartographic map, its global map is shown to be transient, path-dependent and egocentric, inexact and incomplete, and only represents well the last segment of the path traversed. If a large environment is being explored, no single global map that is all-embracing in scope and flexibility is learned. Instead, what could be learned is a series of independent but connected global maps; the extent of each depends on the sensors used and the nature of the environment itself. Albot_1 uses path integration and triangulation for self-localization in each local map and both strategies are commonly used by different species in finding their way (see discussions below).

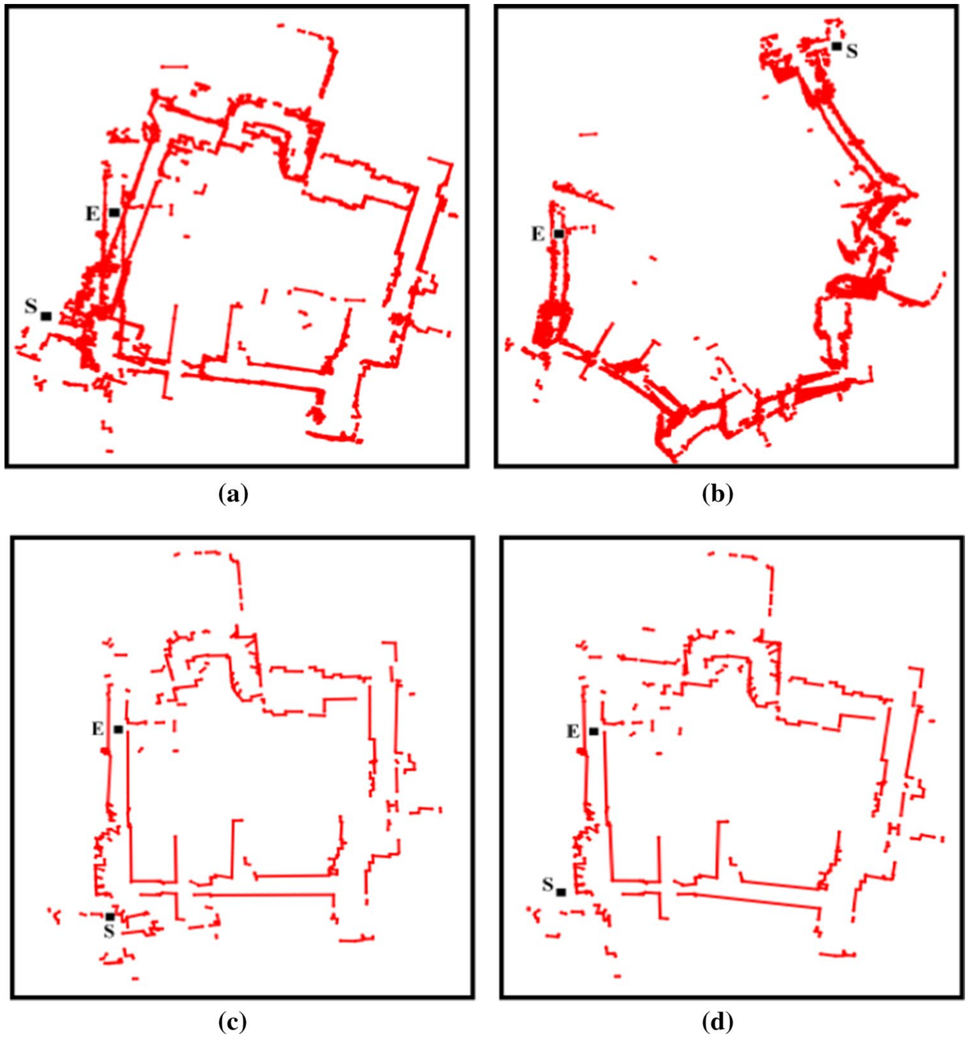

Fig. 16 Maps computed using **a** dataset 1 and **b** dataset 2 by Albot$_x$ and maps computed using **c** dataset 1 and **d** dataset 2 by Albot$_1$

In contrast, the process for computing a global map from integrating successive views requires one to solve a set of problems that include correcting sensor errors, continuous self-localization in the map, continuous updating and recognition of places re-visited. Such a process is shown to be rigid, intolerant to errors, computationally intensive and expensive. This process is a more complex form of path integration whereby one computes the current position of the self from the start and a detailed map of the environment traversed.

General discussion

The mystique of cognitive mapping lies in how and what it computes as a map. Introspectively, the map computed cannot be a complete global, metric map of the environment, so, what kind of a map is computed?

If we assume that a SLAM-based model represents the process that computes a map of one's environment via the integration of successive views, then our experiments using Albot$_x$ indicate that such a process could be too rigid and computationally demanding as a model for cognitive mapping. Cognitive mapping should be highly efficient and adaptive because when exploring the environment one faces numerous other tasks such as attending to what is out there, searching for food, avoiding predators and sometimes, being playful. Our process is modular and view-based. It does not require continuous integration of views, updating of its maps or correction of errors. Without the need to compute a complete map, there is no need to align one's mental space with the physical space prior to getting up and going. One can roam freely and deal with the problems arising without worrying about maintaining a consistent map. Henceforth, we are not in disarray when our mode of travel changes (e.g. from walking to driving) or when we enter into impossible worlds such as those created in virtual environments (e.g. Kluss et al. 2015; Zetzsche et al. 2009). The process degrades gracefully when faced with unexpected changes in the environment. Computationally, our model is thus an attractive model for cognitive mapping.

The nature of the maps computed in our model is telling. In early research on cognitive mapping, one popular idea is that one learns two kinds of maps, a route map and a survey map. The former is meant to describe the physical experience one had while moving through the environment, and the latter is the resulting global view of the environment. Tolman (1948) also discussed that there are two kinds of maps when he introduced the idea of a cognitive map. He described them as a narrow and strip-like map and a broad and comprehensive map. The idea persisted throughout the early research in cognitive mapping (e.g. Shemyakin 1962; Appleyard 1970; Thorndyke and Hays-Roth 1982; O'Keefe and Nadel 1978), but like many other ideas about cognitive mapping, the idea has much intuitive appeal but isn't precise enough to be tested empirically or properly modelled. For example, what distinguishes a map that is narrow and strip-like from a map that is broad and comprehensive? While many researchers consider the latter as some kind of a global metric map, the idea of computing such a map, as noted in the introduction, is problematic.

Our model supports the idea that two maps are computed and they bear close resemblance to some descriptions of these early maps. The route map, according to our model, is a list of local maps that are individual views of the environment taken at different points throughout the journey. It thus affords the sense of being narrow in that the space is bounded by the view and being strip-like in that it is a list portraying the route taken. A survey map is a global metric map but one that is transient, egocentric, inexact and incomplete. In other words, it is nothing like a cartographic map. Our model also predicts that these maps are computed simultaneously (see Ishikawa and Montello 2006; Montello 1998). It does not support the idea that the route map is computed first as a list of landmarks prior to that of a survey map (Siegel and White 1975) or the idea that one's survey map is the result of combining several route maps. Cognitive mapping, according to our model, is best viewed as an independent perceptual process (as opposed to, say, being part of vision) that computes maps that show the bounded space that one is in and not maps that show what is in the environment. One task that clearly distinguishes these two kinds of maps (i.e. one which is bounded and the other unbounded) is the recognition of places re-visited. The maps computed in our model would not facilitate such recognitions and as we have argued, this task is better performed as part of one's conceptual reasoning about the environment.

Without computing a single global map that is all embracing in scope and flexibility, puzzling observations about cognitive mapping raised earlier in the introduction disappear. For example, Beck and Wood's (1976) subjects could not recall that the two physically adjacent structures were adjacent because these structures are, according to our model, remembered at two vastly separated points in their route map. That our perceived distance is affected by the direction of travel can be because what is learned initially is not a single description of the physical path traversed but two separate experiences of the route taken. Consequently, the presence of barriers and intersections, and the time spent in the environment can affect what is learned. Given that the route map keeps growing as one moves about in the environment, some parts of it could be "forgotten". This might account for why subjects draw fragmented sketch maps that show local accuracy only. However, this raises the question of what one could learn as an enduring map of the environment and especially since the survey map is also a transient map. We showed in experiment 2 that one could learn a set of independent but connected global maps, but we did not show how such an enduring map would be used (see future work below). Two important tasks for the latter are recognizing where one is and finding shortcuts when returning home.

Ultimately, our model needs to be tested empirically. It already provides fresh operational definitions (in Feest's (2005) methodological sense) for several of the popular, but controversial/ill-defined, conceptions about cognitive mapping. These include the idea of a route map versus a survey map, an egocentric map versus an allocentric map, a view-based model versus one that integrates successive views and a cartographic-like map versus a cognitive map. With these ideas grounded with a formal model, we hope more rigorous experiments would follow that need not be based on an imaginary process. Our model also leads us to consider two important spatial cognition issues that we will discuss for the remainder of this section. The first concerns the evolution of cognitive mapping. Wang and Spelke (2002) argue that it is an advantage that different species evolve their navigational abilities that build upon a common set of mechanisms. Our experiments using Albot$_1$ show that when computing its maps, the process utilizes functions (such as the tracking of objects in view, path integration and triangulation) that can be found in other perceptual processes. In "On the evolution of cognitive mapping" section, we discuss why our model would support the evolution of cognitive mapping from insects to humans. We also find continuous map updating, as in Albot$_x$'s process, causes the shape of the map to change continuously as a result of correcting errors. In contrast, our process is stable; Albot$_1$'s local map is fixed as it moves in its local environment. In "On perceiving a stable world" section, we discuss why our model would lead to the perception of a stable world.

On the evolution of cognitive mapping

Our model of cognitive mapping describes a process that computes three increasingly sophisticated maps, namely

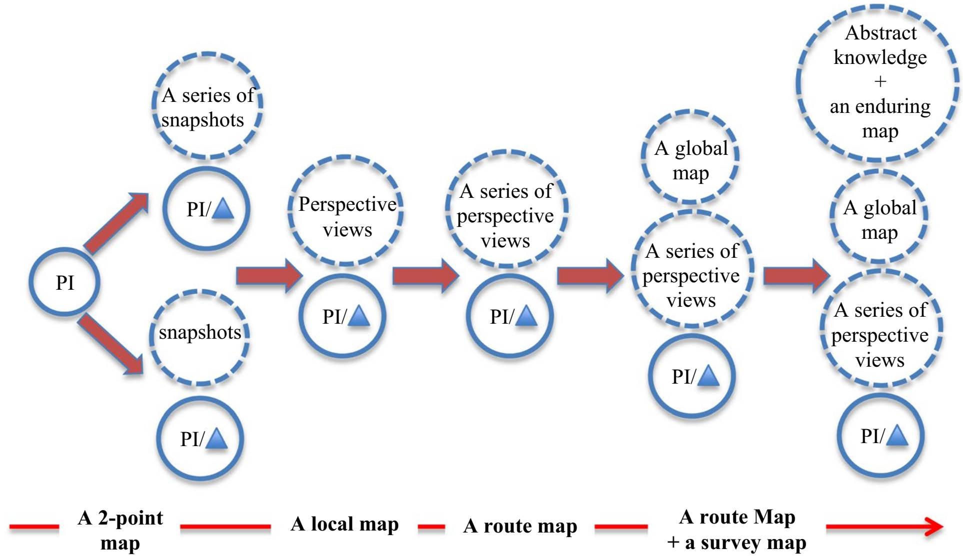

Fig. 17 Tentative evolutionary pathway for cognitive mapping: PI: path integration,: triangulation. Solid circles denote methods used, and dotted circles denote representations computed. Albot$_1$ occupies the penultimate stage (a global map + a series of perspective views)

a local map (i.e. a perspective view), a route map (i.e. a list of overlapping local maps) and a survey map (i.e. an integration of local maps). While most researchers do not consider path integration as part of cognitive mapping, the former produces the simplest form of map, one that displays only the relation between two points in the environment. We argue that the evolution of cognitive mapping begins with the acquisition of a 2-point map (left side of Fig. 17) and then a route map and a survey map (right side of Fig. 17).

It is well known that insects have evolved the use of path integration to travel afar and return home directly and that some correct their inexact 2-point maps by using snapshots to aid their return home (e.g. foragers' "learning walk and flight" (Capaldi and Dyer 1999; Menzel et al. 2000; Nicholson et al. 1999; Muller and Wehner 2010)) and/or to re-calibrate their position in the journey (Collett and Collett 2000). Theoretical models predict that one can re-trace a path using a learned series of snapshots (Gaffin and Brayfield 2016; Smith et al. 2007). Such plausible use of snapshots suggests that a method for triangulating one's position in the environment could have evolved concurrently with a method for path integration. These two methods provide the basic map building tools for cognitive mapping and are also the two basic tools used in our model.

The next step in the process is the evolution of sensors that could deliver a perspective view, thereby allowing one to remember a bounded space prior to moving in it. With visual systems that deliver enhanced depth information, the triangulation method could become the primary method for self-localization. It has the advantage of being independent of one's movement in the environment. Path integration could then be a secondary/redundant source that resets as one moves between local environments, thereby avoiding error accumulation over long distance. Research into the geometric module of the mind (Cheng 1986, 2008) has shown that many species can use the geometry of a place for re-orientation (for reviews of such work, see: Wang and Spelke 2002; Cheng and Newcombe 2005). These findings support the use of a view or a combination of views as a local map. The latter would begin the evolution of a process for computing a global map of the path traversed. Some researchers note that variations in the way different species learn the geometry of a place can be accounted for, in a large part, by the differences in their visual systems (Wystrach and Graham 2012; Cheng et al. 2013). Our model shows that the local map is strictly a product of how one perceives the environment and this, in turn, highlights the importance of what can be delivered by one's visual system.

Our model shows how one could go beyond learning the geometry of a place to learning the geometry of a path, thereby producing a global map of the environment. Our model predicts that the global map emerges as a result of a need to be aware of a much larger environment than what could be perceived in a single view. It is not due to a need to recognize that one is returning to a familiar place. The former task is simpler and therefore it makes evolutionary sense that it is solved first. Note that the map is a transient map and thus some form of an enduring map needs to be learned if one were to use the map to return home using shortcuts, find food items that were hidden earlier and recognize familiar places. The last task would require recognizing familiar objects/landmarks in the environment. For humans as Wang and Spelke (2002) note, we employ symbolic reasoning to learn an even larger "map" of the environment that can be partly metric and partly topological and/or systematically distorted to produce a simplified, but useful map. We can also develop abstract representations of places learned and identify a place using non-physical properties. With such a cognitive mapping process, the empirical question should no longer be whether a map is computed but rather what kind of map is computed and how it is used in dealing with its spatial environment. The former task is limited by the sensors one has and the latter task is limited by one's cognitive capacity.

On perceiving a stable world

For more than a century, psychologists and vision researchers have been debating how we perceive a stable environment as we move (Wallach 1987; O'Regan 1992; Irwin 1996). The debate often focuses on how information is integrated, if at all, between successive views to construct a detailed, yet stable, description of our visual world. One phenomenon, change blindness, has recently caused a stir in the debate because accumulative evidence shows people can fail to notice changes happening between successive views when a disruption occurs at the point that cues a change (for a review of such work, see: Simons and Rensink 2005). If successive views were integrated, it would be impossible, in theory, not to detect changes. Such a logical consequence leads some to conclude that no detailed representation of the visual world is computed (O'Regan 1992; Irwin 1996; O'Regan and Noë 2001). In particular, O'Regan (1992) claims no internal metric-preserving representation needs to be remembered as the outside world can act as our external memory. Others, however, have argued that integration does take place but in limited ways and this affects our ability to detect changes. For example, the resulting representation might not be accessible to conscious scrutiny or that it is not sufficiently detailed or that integration takes place only with information that is in focus (for a review of these works, see: Tatler and Land 2011).

In our model, successive views are not integrated and consequently change blindness does not rule out the presence of a metric-preserving representation as suggested by O'Regan (1992). Rather, such a representation is available in the form of a single perspective view that, in turn, provides a description of the local environment for one to operate in. Change blindness thus demonstrates the failure to attend to information in the current view and highlights a limitation in our object recognition process as opposed to our process for learning the spatial layout of our environment. Note that the former process is independent from the latter. Change blindness provides evidence against continuous integration and not against constructing a description of the visual world. While the world can serve as our external memory, we can only access it one view at a time. We need to integrate these views somehow and without this we cannot point to unseen locations or know how to generate shortcuts (Glennerster et al. 2009).

At the saccadic eye level, a mechanism similar to that used by Albot$_1$, has been proposed to explain how a stable view of the visual world is maintained. Albot$_1$'s novel algorithm tracks reference targets while moving and then uses their positions to locate other objects and its own position in its map. Currie et al. (2000) tested their subjects' ability to detect changes in a picture that is moved in four different ways during a saccade to a target object: "entire picture" shifted, "target object only" shifted, "entire picture except for the target object" shifted and no shift. They find that the position of the target object is most important in detecting intra-saccadic stimulus shifts. In a series of experiments using a blanking effect, Deubel and his colleagues (Deubel 2004; Deubel et al. 2010) find that the target object and in some circumstances those close to it are used to provide a frame of reference for describing the layout of other objects in view after a saccade. They argue this mechanism enables one to perceive a stable world despite saccades.

That humans experience a stable environment while moving in it is clearly demonstrated by an experiment using an immersive virtual environment (Glennerster et al. 2006). In it, the subjects fail to notice the expansion of a room in which they are located even though the room expanded by a factor of four and that other sensory information (including veridical stereo and motion parallax) is available to tell them that the environment has expanded. In discussing their work, Glennerster et al. (2009) concluded that humans do not compute a 3D map via integrating views using a coordinate transformation approach, i.e. an approach similar to Albot$_x$. What they propose is a view-based solution that provides a "co-ordinate frame in which to unite all the information about an object's location" (p. 14). A snapshot model, however, is inadequate since it lacks 3D information while our model is a possible candidate.

Conclusions and future work

We present a model of cognitive mapping that does not compute a single global map of the environment experienced. Instead, it computes, in increasing order of complexity, a point map, a local map, a route map and a series of disjointed global maps. Without computing a single global map, much of the mystery about cognitive mapping disappears. Our model suggests that cognitive mapping would have evolved in different species and would lead to the perception of a stable world. It thus provides a formal model for empirical researchers to continue their investigations into the primacy of cognitive mapping as a process for learning about the environment. Our future work will focus on a different question about cognitive mapping, namely, what could be learned as an enduring map and how it could be used for solving various spatial problems? From this perspective, cognitive researchers have identified an acid test for cognitive mapping, namely how does one generate shortcuts when returning home? Can the next Albot do so too? Can it also recognize where it is in the environment and will it use landmarks in its process? What is a landmark? If we could answer these questions, it would further strengthen our model as a model of cognitive mapping.

Acknowledgements An initial draft of this paper was completed while Professor Yeap was a Fellow at the Hanse-Wissenschaftskolleg at Delmenhorst, Germany. Special thanks to Professor Reto Weiler and Dr. Dorothe Poggel and all the staff at HWK for providing a wonderful environment for research and to Professor Christian Freksa at the University of Bremen for his invitation to spend a sabbatical in Germany. Md. Hossain holds a NZ Government sponsored BuildIT Doctoral Scholarship. We would also like to thank Wenwang Pang (for helping to generate some of the data) and Jordan Alexander, Robin Hankin, John Bancroft and Stephen Hirtle and the numerous anonymous reviewers for helpful comments on earlier drafts. We would also like to acknowledge the various small grants that the Faculty of Design and Creative Technologies at the Auckland University of Technology has provided for this project.

Compliance with ethical standards

Conflicts of interest The authors declare that no conflict of interest.

Appendices: pseudo-codes

Appendix 1

RTSV (V_c , S_t, α, δ) ; Algorithm #1: Recognising Tracked Surfaces in current View

α = angle turned

δ = distance moved

$V_c = \{S_{c1}, S_{c2}, \ldots, S_{cn}\}$; current view consisting of surfaces that satisfy the above three ; criteria (i.e. those that don't, have been removed).

$S_t = \{S_{t1}, S_{t2}, \ldots, S_{tm}\}$; tracked surfaces maintained from previous view

$S'_t \leftarrow \{\}$; tracked surfaces, S_{ti}, that are found in current view, initially empty

Begin

1: **for** each S_{ti} in S_t **do** $S'_t \leftarrow S'_t$ + transform (S_{ti}, α, δ)

2: $S_t \leftarrow \{\}$; assume no tracked surfaces are still in view

3: **for** each S'_{ti} in S'_t **do** ; decide which tracked surfaces are still in view

4:     **for** each S_{ci} in V_c **do**

5:         **if (**angle(S'_{ti}, S_{ci})<5^o) **&** (dist(between occluding points) < 40cm) **then** $S_t \leftarrow S_t + S_{ci}$

6:     **endfor**

7: **endfor**

8: **return** S_t ; a new set of tracked surfaces that are still in view

End

Appendix 2

Main () ; Algorithm #2

Initialisation

$V_c = \{S_1, S_2, \ldots, S_m\}$; view at the start

$R^c_{0,0}$ = the robot's current position

$M_R = \{(V_c, R^c_{x,y})\}$; the route map initialised with the starting view

$M_G = V_c$; the global map initialised with the starting view

Use-V_c $\leftarrow$ false ; to prevent looping

Begin

1: $S_t \leftarrow$ find all trackable surfaces in V_c ; $S_t = \{S_{t1}, S_{t2}, \ldots S_{tn}\}$

2: $V_{c\text{-}1} \leftarrow V_c$

3: $V_c \leftarrow$ Get_a_new_view(α, δ) ; α = angle turned, δ = dist moved

4: $S_t \leftarrow$ RTSV(V_c, S_t, α, δ) ; algorithm #1 – find which tracked surfaces still remain in view

5: **When** $|S_t| < 4$ **do** ;; $|S_t|$ means number of surfaces

6: $R^{c\text{-}1}_{x,y} \leftarrow$ triangulate($V_{c\text{-}1}$, S_t) ; find robot's position in current local map

7: update-maps(if Use-V_c then V_c else $V_{c\text{-}1}$, $R^{c\text{-}1}_{x,y}$) ; see algorithm #3

8: **If** $|S_t| = 0$ **then**

9: move-using-PI ($R^{c\text{-}1}_{x,y}$, α, δ) ; shift to PI mode - see algorithm #4

10: **Goto** 3

11: **else If** $|S_t| < 4$ **do** Use-V_c $\leftarrow$ true **endif** ;; expect updating again

12: **Goto** 4 ; continue to explore using views

13: **endif**

14: **endwhen**

15: Use-V_c $\leftarrow$ false

16: **Goto 2**

End

Update-maps (V,R) ; Algorithm #3: Add a local map to the route map & to the global map

V = current view

R = position of robot in the current local map

Begin

1: $M_R \leftarrow M_R + (V, R)$; add the next local map to the route map

2: $S_t \leftarrow$ find all trackable surfaces in V

3: V' $\leftarrow$ transform V into M_G co-ordinates using R

4: $M_G \leftarrow$ Update-Global-Map(M_G, V', R) ;; see algorithm #5

End

Move-using-PI (R, α, δ) **; Algorithm #4: Updating map using path integration**

α = angle turned

δ = distance moved

R = robot's position in the current local map.

Begin

1: $R^c_{x,y}$ ← compute robot's current position after turning α and move δ from R

2: update-maps(V_c, $R^c_{x,y}$)

3: V_{c-1} ← V_c

4: V_c ← Get_a_new_view(α', δ') ; α' = new angle turned, δ' = new dist moved

5: S_t ← find all trackable surfaces in V_c

6: $R^c_{x,y}$ ← compute robot's new position as it turns α and move δ from $R^c_{x,y}$

7: **If** $S_t > 3$ **then** ; the new view is full of trackable surfaces

8: update-maps(V_c, $R^c_{x,y}$) ; use this view as a local map

9: **return** ; to move using view navigation

10: **else Goto 2** ; still moving in a "featureless" environment

11: **endif**

12: use-V_c ← true ;; just in case the next move causes an update

End

Appendix 3

Update-Global-Map (M_G, M, $R_{x,y}$):

$M = \{S_1, S_2, \ldots, S_m\}$; incoming local map in M_G's co-ordinates

$M_G = \{S'_1, S'_2, \ldots, S'_n\}$; the global map

$R_{x,y}$ = robot's position in M_G

Begin {algorithm #5: Updating the Global Map}

Compute a polygon, P, by joining the edge points of adjacent surfaces in M when perceived in a clockwise direction

P ← Enlarge P by 50cm

M_G' ← M_G

Until M_G' = {} **do** ; find all affected maps

    if S'_i in M_G' is inside or intersects P **then**

        if map(S'_i)=current **then** delete(S'_i in M_G & in M_G')

        else maps-affected ← maps-affected + map(S'_i)

            M_G' ← M_G' – all-surfaces-of-map(S'_i) **end if**

    end if

end until

;; Expand calculates the group of local maps affected as discussed in section 3.3

maps-affected ← Expand(maps-affected)

M_G' ← M_G

forall S'_i in M_G' **do**

    if (map(S'_i) in maps-affected) **and** (S'_i lies on the facing side of $Albot_1$ at its position $R_{x,y}$)

        then M_G ← M_G - S'_i

    end if

end forall

M_G ← M_G + M ;; add incoming map to it after deleting overlapping information

Return M_G

End

References

Appleyard D (1970) Styles and methods of structuring a city. Environ Behav 2:100–117

Bailey T, Durrant-Whyte H (2006) Simultaneous localization and mapping (SLAM): Part II. IEEE Robotics Automation Mag 13(3):108–117

Barrett HC, Kurzban R (2006) Modularity in cognition: framing the debate. Psychol Rev 111(3):628–647

Beck RJ, Wood D (1976) Cognitive transformation of information from urban geographic fields to mental maps. Environ Behav 8:199–238

Benhamou S (1996) No evidence for cognitive mapping in rats. Anim Behav 52:201–212

Bennett ATD (1996) Do animals have cognitive maps? J Exp Biol 199:219–224

Brown MF (1992) Does a cognitive map guide choices in the radial-arm maze? J Exp Psychol Anim Behav Process 18(1):56–66

Buchner S, Jansen-Osmann P (2008) Is route learning more than serial learning? Spatial Cogn Comp 8:289–305

Burgess N (2006) Spatial memory: how egocentric and allocentric combine. Trends in Cogn Sc 10:551–557

Capaldi EA, Dyer FC (1999) The role of orientation flights on homing performance in honeybees. J Exp Biol 202:1655–1666

Cartwright BA, Collett TS (1982) How honey bees use landmarks to guide their return to a food source. Nature 295:560–564

Cheeseman JF, Millar CD, Greggers U, Lehmann K, Pawley MDM, Gallistel CR, Warman GR, Menzel R (2014) Reply to Cheung et al: The cognitive map hypothesis remains the best interpretation of the data in honeybee navigation. Proc Natl Acad Sci USA 111(42):E4398

Cheng K (1986) A purely geometric module in the rat's spatial representation. Cognition 23:162–171

Cheng K (2008) Whither geometry? Troubles of the geometric module. Trends in Cogn Sci 12(9):355–361

Cheng K, Newcombe NS (2005) Is there a geometric module for spatial orientation? Squaring theory and evidence. Psychonom Bull Rev 12(1):1–23

Cheng K, Huttenlocher J, Newcombe NS (2013) 25 years of research on the use of geometry in spatial reorientation: a current theoretical perspective. Psychonom Bull Rev 20:1033–1054

Cheung A, Stürzl W, Zeil J (2008) The information content of panoramic images II: view-based navigation in nonrectangular experimental arenas. J Exp Psychol: Anim Behav Processes 34(1):15–30

Cheung A, Collett M, Collett TS, Dewar A, Dyer F, Graham P, Mangan M, Narendra A, Philippides A, Stürzl W, Webb B, Wystrach A, Zeil J (2014) Still no convincing evidence for cognitive map use by honeybees. Proc Natl Acad Sc USA 111(42):E4396–E4397

Cohen J, Bussey K (2003) Rats form cognitive maps from spatial configurations of proximal arm cues in an enclosed 4-arm radial maze. Learn Motiv 34:168–184

Collett M, Collett TS (2000) How do insects use path integration for their navigation? Biol Cybernetics 83:245–259

Crompton A (2005) Perceived distance in the city as a function of time. Env Behav 38(2):173–182

Currie C, McConkie GW, Carlson-Radvansky LA, Irwin DE (2000) The role of the saccade target object in the perception of a visually stable world. Percep Psychophys 62(4):673–683

Deubel H (2004) Localization of targets across saccades: role of landmark objects. Vis Cogn 11(2/3):173–202

Deubel H, Koch C, Bridgeman B (2010) Landmarks facilitate visual space constancy across saccades and during fixation. Vis Res 50:249–259

Downs RM, Stea D (1973) Image and environment: Cognitive mapping and spatial behavior. Aldine, Chicago

Durrant-Whyte H, Bailey T (2006) Simultaneous localization and mapping: Part I. IEEE Robot Autom Mag 13(2):99–110

Ekstrom AD, Arnold AE, Laria G (2014) A critical review of the allocentric spatial representation and its neural underpinnings: toward a network-based perspective. Front Hum Neurosci 8:803

Etienne AS, Berlie J, Georgakopoulos J, Maurer R (1998) Role of dead reckoning in navigation. In: Healey S (ed) Spatial representation in animals. Oxford University Press, New York, pp 54–68

Evans GW (1980) Environmental cognition. Psychol Bull 88(2):259–287

Evans G (1982) The varieties of reference. Oxford University Press, Oxford

Feest U (2005) Operationism in psychology: what the debate is about, what the debate should be about. J History Behav Sci 41(2):131–149

Fodor J (1983) The modularity of mind. MIT Press, Cambridge

Gaffin DD, Brayfield BP (2016) Autonomous visual navigation of an indoor environment using a parsimonious, insect inspired familiarity algorithm. PLoS ONE 11(4):e0153706

Gallistel CR (1990) The organisation of learning. MIT Press, Cambridge

Glennerster A, Tcheang L, Gilson SJ, Fitzgibbon AW, Parker AJ (2006) Humans ignore motion and stereo cues in favour of a fictional stable world. Curr Biol 16:428–443

Glennerster A, Hansard ME, Fitzgibbon AW (2009) View-based approaches to spatial representation in human vision. In: Cremers D, Rosenhahn B, Yuille AL, Schmidt FR (eds) Statistical and geometrical approaches to visual motion analysis, vol 5604. Lecture Notes in Computer Science. Springer, Berlin, pp 193–208

Golledge RG, Smith TR, Pellegrino JW, Doherty S, Marshall SP (1985) A conceptual model and empirical analysis of children's acquisition of spatial knowledge. J Environ Psychol 5:125–152

Graham P, Collett TS (2002) View-based navigation in insects: how wood ants (*Formica rufa* L.) look at and are guided by extended landmarks. J Exp Biol 205:2499–2509

Grisetti G, Stachniss C, Burgard W (2007) Improved techniques for grid mapping with Rao-Blackwellized particle filters. IEEE Trans Robotics 23(1):34–46

Howard A, Roy N (2003) The robotics data set preository (Radish). http://radish.sourceforge.net/

Irwin DE (1996) Integrating information across saccadic eye movements. Curr Dir Psychological Sci 5(4):94–100

Ishikawa T, Montello DR (2006) Spatial knowledge acquisition from direct experience in the environment: individual differences in the development of metric knowledge and the integration of separately learned places. Cogn Psychol 52:93–129

Jeffery K, Burgess N (2006) A metric for the cognitive map: found at last? Trends Cogn Sci 10(1):1–3

Kluss T, Marsh WE, Zetzsche C, Schill K (2015) Representation of impossible worlds in the cognitive map. Cognit Process 16(Suppl 1):S271–S276

Koriat A, Goldsmith M, Pansky A (2000) Toward a psychology of memory accuracy. Ann Rev Psychol 51:481–537

Kuipers B, Byun YT (1991) A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. Robot Auton Syst 8:47–63

Luhrs M, Dammhahn M, Kappeler PM, Fichtel C (2009) Spatial memory in the grey mouse lemur. Anim Cogn 12:599–609

Lynch K (1960) The image of the city. MIT Press, Cambridge

Mackintosh NJ (2002) Do not ask whether they have a cognitive map, but how they find their way about. Psicologica 23:165–185

Mandelbaum E (2013) Numerical architecture. Topics. Cogn Sci 5:367–386

Mandelbaum E (2015) The automatic and the ballistic: modularity beyond perceptual processes. Philos Psychol 28(8):1147–1156. https://doi.org/10.1080/09515089.2014.950217

McDonald TP, Pellegrino JW (1993) Psychological perspectives on spatial cognition. In: Garling T, Golledge RG (eds) Behavior and environment: Psychological and geographical approaches. Elsevier Science Publisher, Amsterdam, pp 47–82

McNamara TP (2003) How are the locations of objects in the environment represented in memory? In: Freksa C, Brauer W, Habel C, Wender K (eds) Spatial Cognition III: Routes and navigation, human memory and learning, spatial representation and spatial reasoning. Springer, Berlin, pp 174–191

Meilinger T, Vosgerau G (2010) Putting egocentric and allocentric into perspective. In: Hölscher C et al (eds) Spatial Cognition VII, LNAI 6222. Springer, Berlin, pp 207–221

Menzel R, Brandt R, Gumbert A, Komischke B, Kunze J (2000) Two spatial memories for honeybee navigation. Proc R Soc London B 267:961–968

Montello DR (1997) The perception and cognition of environmental distance: Direct sources of information. In: Hirtle SC, Frank AU (eds) Spatial information theory: A theoretical basis for GIS. Springer, Berlin, pp 297–311

Montello DR (1998) A new framework for understanding the acquisition of spatial knowledge in large-scale environments. In: Egenhofer MJ, Golledge RG (eds) Spatial and temporal reasoning in geographic information systems. Oxford University Press, New York, pp 143–154

Mou W, McNamara TP, Valiquette CM, Rump B (2004) Allocentric and egocentric updating of spatial memories. J Exp Psychol Learn Mem Cogn 30(1):142–157

Muller M, Wehner R (2010) Path integration provides a scaffold for landmark learning in desert ants. Curr Biol 20:1368–1371

Nicholson DJ, Judd SP, Cartwright BA, Collett TS (1999) Learning walks and landmark guidance in wood ants (formica rufa). J Exp Psychol 202:1831–1838

Normand E, Boesch C (2009) Sophisticated Euclidean maps in forest chimpanzees. Anim Behav 77:1195–1201

O'Keefe J, Nadel L (1978) The hippocampus as a cognitive map. Clarendon Press, Oxford

O'Regan JK (1992) Solving the "real" mysteries of visual perception: the world as an outside memory. Can J Psychol 46:461–488

O'Regan JK, Noë A (2001) A sensorimotor account of vision and visual consciousness. Behav Brain Sci 24:939–1011

Pecchia T, Vollortigara G (2010) View-based strategy for reorientation by geometry. J Exp Biol 213:2987–2996

Polansky L, Kilian W, Wittemyer G (2015) Elucidating the significance of spatial memory on movement decisions by African savannah elephants using state-space models. Proc R Soc London B 282:2014–3042

Raghubir P, Morwitz VG, Chakravarti A (2011) Spatial categorization and time perception: why does it take less time to get home? J Consumer Psychol 21(2):192–198

Rovine MJ, Weisman GD (1989) Sketch-map variables as predictors of way-finding performance. J Environ Psychol 9:217–232

Rump B, McNamara TP (2007) Updating in models of spatial memory. In: Barkowsky T, Knauff M, Montello DR (eds) Spatial cognition V. Springer, Berlin, pp 249–269

Schölkopf B, Mallot HA (1995) View-based cognitive mapping and path planning. Adapt Behav 3:311–348

Scholl BJ (2007) Object persistence in philosophy and psychology. Mind Lang 22(5):563–591

Shemyakin FN (1962) General problems of orientation in space and space representations. In: Ananyev BG (ed) Psychological science in the USSR (volume 1). Office of Technical Reports, Washington D.C. 62-11083

Sholl MJ (2001) The role of a self-reference system in spatial navigation. In: Montello DR (ed) Spatial information theory: Foundations of geographical information science. Springer, Berlin, pp 217–232

Siegel AW, White SH (1975) The development of spatial representations of large- scale environments. Adv. Child Dev. Behav by H.W. Reese 10: 9–55

Simons DJ, Rensink RA (2005) Change blindness: past, present, and future. Trends Cogn Sci 9(1):78–97

Smith L, Philippides A, Graham P, Baddeley B, Husbands P (2007) Linked local navigation for visual route guidance. Adapt Behav 15(3):257–271

Tatler BW, Land MF (2011) Vision and the representation of the surroundings in spatial memory. Philos Trans R Soc London B 366:596–610

Thorndyke PW, Hays-Roth B (1982) Differences in spatial knowledge acquired from maps and navigation. Cogn Psychol 14:560–589

Thrun S, Fox D, Burgard W, Dellaert F (2001) Robust Monte Carlo localization for mobile robots. Art Intell 128(1–2):99–141

Tolman EC (1948) Cognitive maps in rats and men. Psychol Rev 55:189–208

Tversky B (1993) Cognitive maps, cognitive collages, and spatial mental models. In: Frank AU, Campari I (eds) Spatial information theory: A theoretical basis for GIS. Springer, New York, pp 14–24

Wagner T, Visser U, Herzog O (2004) Egocentric qualitative spatial knowledge representation for physical robots. Robot Auton Syst 49:25–42

Wallach H (1987) Perceiving a stable environment when one moves. A Rev Psychol 38:1–27

Wang RF, Spelke ES (2002) Human spatial representation: insights from animals. Trends Cogn Sci 6(9):376–382

Wystrach A, Graham P (2012) View-based matching can be more than image matching: the importance of considering an animal's perspective. i-Perception 3:547–549

Yeap WK (2011) How Albot$_0$ finds its way home: a novel approach to cognitive mapping using robots. Topics Cogn Sci 3(4):707–721

Yeap WK (2014) On egocentric and allocentric maps. In: Freksa C et al. (eds), Spatial Cognition IX, LNAI 8684. Springer, Berlin, 62–75

Zetzsche C, Wolter J, Galbraith C, Schill K (2009) Representation of space: image-like or sensorimotor? Spatial Vis 22(5):409–424

Zhao M, Warren WH (2015) Environmental stability modulates the role of path integration in human navigation. Cognition 142:96–109