# Sensory Anticipation for Autonomous Selection of Robot Landmarks

Jason Fleischer[1], Stephen Marsland[2], and Jonathan Shapiro[1]

[1] Department of Computer Science
University of Manchester
Manchester M13 9PL, UK
{jgf,jls}@cs.man.ac.uk
[2] Imaging Science and Biomedical Engineering
Stopford Building
University of Manchester
Manchester M13 9PT, UK
stephen.marsland@man.ac.uk

**Abstract.** There are many ways to define what constitutes a suitable landmark for mobile robot navigation, and automatically extracting landmarks from an environment as the robot travels is an open research problem. This paper describes an automatic landmark selection algorithm that chooses as landmarks any places where a trained sensory anticipation model makes poor predictions. The model is applied to a route navigation task, and the results are evaluated according to how well landmarks align between different runs on the same route. The quality of landmark matches is compared for several types of sensory anticipation models and also against a non-anticipatory landmark selector. We extend and correct the analysis presented in [6] and also present a more complete picture of the importance of sensory anticipation to the landmark selection process. Finally, we show that the system can navigate reliably in a goal-oriented route-following task, and we compare success rates using only metric distances with using a combination of odometric and landmark category information.

## 1 Introduction

Landmarks are commonly used for mobile robot navigation, but there is little consensus about the definition of the term and many theories about what constitutes a good landmark. In this paper we consider *perceptual* landmarks, for which we take the usual definition – a robot's sensor values in a particular location. A good perceptual landmark is one that enables a robot to maintain an accurate estimate of its location while traversing an area that it has previously mapped. Landmark-based localisation in this sense forms the basis of many successful mobile robot navigation systems, see for example [5, 20].

The question of how to select landmarks, then, is an important one, yet it is often solved by avoiding the issue, simply taking sensory scans at regular intervals, and taking every scan as a landmark. In many environments a large number of perceptions are almost identical – especially to the limited sensors available to a robot – so the majority of these landmarks do not aid navigation. A slightly better approach is to *select* landmarks in some way, for example by having an experimenter choose landmarks while looking at images taken by the robot [11], or by using some form of statistical learning, e.g., [17, 3].

In this paper we address the problem of learning to select landmarks so that a robot can reliably navigate a route without getting lost. The robot should select landmarks that will be perceivable each time it follows the route (consistent), easily distinguishable from nearby locations (distinctive), frequent enough to be useful for localisation, but not so frequent that localisation is computationally expensive. These landmarks are not required to be completely unique, so that a robot might know exactly where it is from single perception – this may well be impossible in many environments. We assume that a robot is following a route and must localise itself as it travels, rather than trying to localise from scratch against all possible locations.

One solution to this problem, described in section 2, is to learn a model of sensory anticipations in a typical operating environment. This model of normal sensory sequences is used to predict the robot's next perception at every time step, and a landmark is defined to be any place where the robot's anticipations are not accurate; in other words, those places that are not adequately described in the model. This is a form of novelty detection. A Kalman filter is used both to remove the effects of noise in the predictions and to maintain statistics used for the selection of landmarks. Landmarks are therefore based on the novelty of the current perception in relation to recent context, ensuring that they are distinctive compared to nearby perceptions. The landmarks are consistent because the Kalman filter helps to remove both the effects of robot path variation and small environmental changes that may not be detected every time a route is followed.

We also investigate an alternative to anticipatory landmark detection, based on perceptual novelty in relation to the entire perceptual history. We use a growing neural network that adds nodes if a perception is not well described by previous perceptual categories. The growing network is trained in the typical operating environment, in the same way as the anticipatory method, and then operated in the new environment, with the growth mechanism frozen so that new nodes cannot be added. Any location where the network would have added a new node during training is considered to be a landmark. This approach to landmark detection, described in section 3, is completely different; it has no mechanisms in common with the anticipatory method, landmarks are selected based on their similarity to all perceptions seen previously, without any information about current perceptions. The perceptual novelty approach selects a landmark because it is distinctive with respect to the robot's entire perceptual experience.

The drawback of the anticipatory landmark detector is that an assumption must be made that the model's learning ability is sufficient to predict well, but not so great that it never produces enough error to detect a landmark. Previous experience [14] has shown that single-layer neural networks of various types meet this assumption in office corridor environments. A drawback of both approaches is that they require selecting parameters that determine the amount of novelty a perception must posses before it is selected as a landmark.

In previous publications, the use of sensory predictions to automatically select landmarks in static straight corridors where the robot took sensor readings at fixed intervals has been investigated [14]. The anticipatory method has also been applied to robot navigation in more complex routes involving turns, and some simple methods of localisation based on following route descriptions was investigated [6]. The perceptual novelty method has never before been applied as a landmark detector for localisation.

This paper extends and corrects the previous analysis in [6] that investigated the performance of various landmark alignment methods and parameter values for the anticipatory landmark detector. Those analyses were based on an assumption that the scores were normally distributed, which has since been found to be false in certain cases. In addition, we utilised many pairs of $t$-tests, opening up the possibility of an erroneous conclusion occurring due to the sheer number of comparisons being made. This paper verifies the previous results with distribution-free ANOVA analyses [7] in sections 4.3 and 4.4. We also look at the use of different complexities of sensory anticipation model and the resulting differences in landmark alignment in section 4.5. We compare the perceptual novelty method of landmark detection to the sensory anticipation method in terms of landmark alignments in section 4.6, and we present the results of using automatically selected landmarks in a route-following task in an office corridor environment in section 4.7. Finally, we close the paper with a discussion (section 5) and conclusions (section 6).

## 2 Landmark Selection by Sensory Anticipation

Anticipations are predictions of future values that have an effect on the current behaviour of the agent. There are several types of anticipatory mechanisms [4]; the direct prediction of sensory values given recent perceptions is one of them. Sensory predictions can be used either passively, as an attentional mechanism, or actively, in a sensorimotor context.

In neuromotor control models, sensory predictions are generated from current sensory values by feeding the motor commands to be executed through a forward model of the system's kinematics. The system can therefore detect the difference between sensory changes induced by self-motion and those that occur due to external factors, and cancel those effects if necessary [19], a process known as reafference. An example robotic application of sensorimotor prediction is the use of predictions to increase the accuracy of optic flow measurements in an active perception framework [16]. A reafferent model has also been used to investigate cognitive models of sensory anticipation, investigating the possibility that internal simulation can take the place of sensory inputs, allowing the robot to act blindly for short periods of time [8].

Our approach is to use sensory anticipations passively, we do not attempt to correct perceptions given self-movement, as in the reafferent case. The anticipatory mechanism described in this paper can be viewed as focusing the attention of the robot onto a particular perception in the temporal sequence because of its novelty with respect to the recent perceptions. This is related to the topics of change detection and novelty detection [12, 13].

The landmark detection system consists of a sensory anticipation network and a method of detecting when the difference between the prediction of the next sensor values and the actual perceptions (the error) is large enough to be considered a landmark. The sensory anticipation model used in this paper is a single-layer artificial neural network, as previous work [14] has shown this to be a suitable method. The network is trained in a latent learning fashion, by allowing the robot to travel through the environment using a wall-following behaviour, collecting sensor data and training its anticipatory model, as described in section 2.1. Once trained, the error between the model's predictions and the real perceptions is used to select landmarks. The prediction error is sent through a Kalman filter to both remove the effects of noise and to use the estimated statistics to select landmarks. Because the Kalman filter is recursively updated and includes a predictive noise model it provides the estimation of context perhaps nearly as much as the sensory anticipation model. Finally, each landmark is categorised using a Self-Organising Map, as described in section 2.2.

### 2.1 The Sensory Anticipation Model

Our sensory anticipation model learns to predict the next values of the five sonar sensors that face the wall being followed by the robot. The side-facing sonars are preferable because (a) they do not register humans walking along the corridor, and (b) for route navigation in a corridor potentially interesting landmarks are usually doors and hallway branches that will be best detected when directly to the side of the robot. The following types of single-layer artificial neural networks are investigated for learning the sensory anticipation model:

**Standard** Each node represents a sonar value, current (normalised) sensor readings being used as inputs, with the network attempting to predict the sensor readings at the next time-step. The output nodes have a sigmoidal activation function and are connected to one or more input nodes and a bias input. The output layer is not fully connected because sonars to the rear of the robot have no use in predicting future values of more forward sensors. Thus, each output node is only connected to its own input node and input nodes that represent sonars further forward on the robot; see [14] for more details.

**Lagged** This network uses the same structure as the standard neural network, but the inputs values consist of normalised sonar values from both the current time and the previous time-step.

**Recurrent** This is similar to the lagged version, except that the inputs consist of the current sonar inputs and the network outputs at the previous time-step.

**Naïve** This is not a neural network, but merely assumes that the next perception will be identical to the current one.

These models are compared with each other for the landmark selection task in section 4.5 and with a non-anticipatory on-line landmark detection method in section 4.6.

The neural network is trained to acquire a model of the relationship between successive sensory perceptions by adapting the weights during a training phase, where the robot explores an environment and records its sensory perceptions. At each time-step the current sensory perceptions are used as network inputs and the previous sensory perceptions are used as training examples. The weights are adapted by a least-mean-squares learning rule and trained for 17 epochs on the data with a learning rate of 0.2. The number of epochs was determined by using early stopping on several sample datasets and picking a reasonable value for general use.

We reiterate that this model makes no attempt at performing reafference; it learns to minimise prediction error including robot motion. Instead, we employ a Kalman filter [9] to remove components of the prediction error signal that are due to robot orientation and sensor repeatability, and to provide a systematic method to determine when sensory predictions are sufficiently incorrect to be labelled as a landmark. More detailed discussions of our implementation can be found in [6, 14].

The Kalman filter is a method of recursively estimating the state of a discrete-time controlled process that is governed by a linear stochastic difference equation. In our case the filter estimates a single scalar variable, the sum-squared error of the prediction network, $E$, which would be a constant value, hopefully near zero, if the robot maintained perfect orientation to a perfectly even wall. However, there is noise in $E$ from both repeatability (measurement noise) and robot orientation and position (process noise). The Kalman filter attempts to optimally re-estimate $E$ at each step so as to remove the effects of both measurement and process noise, leaving behind only the prediction error actually produced by variation of the environment. The filter equations use a constant measurement noise model and a time-varying process noise model to compute a gain, $K$, that is in turn used to recursively update estimates of the true prediction error, $\widehat{E}$, and its variance, $V$. The variance estimate can be used to select as a landmark any perception where $\widehat{E}(t) > \bar{E}(t) + n\sqrt{V(t)}$. The parameter $n$ provides a way of adjusting the required level of landmark distinctiveness, and therefore to some extent landmark frequency. Typically, we find that values of $n$ between three and five work well.

### 2.2 Landmark Categorisation

Some of the localisation methods presented in section 4.3 will not align the current perception with a location in the route description unless they have the same landmark category. Categorisation in this sense is merely clustering together similar robot sensor perceptions. We used Kohonen's Self-Organizing

Map (SOM) [10], an unsupervised neural network based on vector quantisation, to perform categorisation of variance normalised sonar sensor values from the robot. The SOM is trained by comparing each sensor input vector to every node in the network and selecting the best-matching node, in terms of Euclidean distance between the input and the node weights, as the winner. The winning node and other nodes nearby in the output topology are adjusted so that their weights are closer to the current input. The category of a perception is the identity of the winning node in the map for that perception.

We used the SOM Toolbox [18] to generate the SOMs used in this paper. The SOMs all have $5 \times 4$ toroidal network topology, and the toolbox was allowed to automatically select the training parameters, based on the number of inputs, the number of data points, and the principal components of the training set. There are several possible choices of SOM inputs – we investigated using either all 16 sonar sensors or only those 5 facing the wall being followed – the same sonars used for landmark selection (see section 2.1). We also tried training the SOM on every perception in the training environment, or just on those that are classified as landmarks. The various types of SOM are compared in section 4.4.

## 3 Landmark Selection Using Perceptual Novelty

In machine learning a novelty filter is an algorithm that detects when inputs differ significantly in some way from those that it has seen before. This is different to the standard classification problem, where each new perception is classified as belonging to a known category after the algorithm has been trained on many examples of all the different classes. We used a novelty filter known as the Grow When Required (GWR) network [15] (see also the similar FOSART algorithm [2, 1]) to learn a model of a training environment. The GWR network is a self-organising topology-preserving network that can dynamically create and destroy network nodes, allowing it to model dynamic datasets on-line. When the robot explores a test environment, any perceptions that the novelty detector highlights as novel inputs were labelled as landmarks.

Learning in the GWR network is driven by an insertion threshold $0 \leq a_T \leq 1$, which can be thought of as tunable generalisation; the amount to which the network generalises between similar perceptions is controlled by the amount of discrepancy between perceptions that triggers a new node. Lower values of the insertion threshold produce fewer, more general categories. In this paper, the network is trained on one run in a subset of the hallways that are used as the testing environment. Thereafter, the network is fixed, and if the network would normally add a new node then the input is considered novel, and therefore a landmark. Experiments with the GWR network can be found in section 4.6.

## 4 Experiments

### 4.1 Physical Setup

The experiments in this paper were performed on a Nomad Scout, a differential drive robot with 16 Polaroid sonar sensors capable of giving range information on objects between 15 cm and 6 m away from the robot. The sensor values were updated as quickly as the processing speed of the PC controller would allow, giving less temporal structure to the data than there would be if the sensors were only updated at fixed distance or time intervals. Each experimental run consisted of between 30 and 200 metres of travel in normal office building corridors that were not modified for the robot's use in any way. These runs were made during daytime hours with normal use being made of the corridors; no attempt was made to remove data anomalies due to people walking through the sensor range.

The robot used a hard-wired wall-following program to follow the left-hand wall at a constant distance. Noise in the sonar sensor values comes principally from the imperfect wall-follower, but also from both systematic and repeatability errors in the sensors themselves. Another source of noise is the odometry, which is used to help localise the robot. A rule of thumb for our robot type is that maximum odometry drift is roughly five percent of distance travelled. Therefore we assume that drift can be modelled as $\nu(d) = 0.25 + 0.05d$, where $\nu$ is the maximum possible error and $d$ is distance travelled in metres.

In the landmark alignment experiments (sections 4.3–4.5) three different, adjacent, hallway sections were used for training the landmark detection algorithms. The robot travelled through each hallway three times, thus producing a total of nine different datasets for training the algorithms. The training data were used to train both the landmark detector and the landmark categoriser. The test data were produced by propping open the doors between the three hallways and collecting a further three data runs using a 200 m long route through all three hallways. To compare the alignment between landmarks the three test sets were paired in all six possible permutations (since order of comparison matters) and the measures described in the next section were calculated. In all cases training and testing were performed off-line, using the data collected by the robot.

### 4.2 Analysis Methods

We use four metrics to quantitatively evaluate landmark alignments, and therefore localisation quality, between two runs of the robot along the same route:

**Category score** The number of times that the aligned landmarks share the same category in the SOM, divided by the number of alignments.

**Distance score** The mean of $\exp \frac{-\delta_k^2}{2\nu_k^2}$ for all aligned landmarks, where $\delta_k$ is the difference between the recorded odometry distance for landmark $k$ and the distance of its aligned partner, and $\nu_k$ is the possible odometric drift at that landmark.

**Sequence score** The fraction of alignments where the aligned partner of landmark $k+1$ is the same distance, or further on, from the aligned partner of landmark $k$. Landmarks without assigned alignments are not counted as a break in the sequence.

**Percentage of landmark alignments** The number of alignments made over the total number of landmarks in the current run.

Note that the first three metrics do not penalise alignments that produce fewer matches of better quality. This is based on the assumption that navigation with many landmarks of uncertain quality is more difficult than navigation with fewer, better quality landmarks, i.e., landmarks that are found consistently. It is also worth noting that these metrics give only an approximation to the accuracy of the landmark matching. There is no way to ensure that matching landmarks really are in the same location, since there is no global metric localisation system such as differential GPS or external cameras available. The best possible evaluation of the alignments in these circumstances is the success rate of using the alignments to find a goal location, as shown in section 4.7.

These scores are evaluated using distribution-free statistical tests of parameters, see for example [7]. In particular, we use a Kruskal-Wallis ANOVA, which tests if the null hypothesis – that all choices of treatments (i.e., experimental parameters) are equal – is false. We also use a Freidman ANOVA, which does the same thing, but also seeks to cancel out the effects of other systematic differences by blocking together trials that are homogeneous with respect to other factors that might influence the results. If a difference does exist, then a Tukey-Kramer multiple comparison of means is used to try grouping together those treatments that are not statistically distinguishable at some confidence level, therefore allowing us to determine the best performing treatments.

### 4.3 Aligning Route Descriptions

In this section, we propose some simple methods of robot localisation on a route, and compare them to select the best one to use for subsequent experiments on landmark detection. We find that the best results come from the CSL alignment method, which uses a combination of landmark category, distance travelled, and an odometric error model. CSL not only produces some of the best scores, but just as importantly it produces alignments between runs that are qualitatively pleasing.

Each trip that a robot makes along a route is recorded in a combined metric and topological route description, consisting of the sequence of landmarks perceived along the route, along with the distances between them as measured by the robot's internal odometry. The perceptual category of each landmark generated by the SOM is also recorded. When a robot attempts to follow a route it has previously taken it attempts to match it's current landmark perception with the stored route description. We investigated six different methods of producing such alignments, reflecting the possible combinations of information stored in the route description (sequence, distance, and category) and an odometric error model. The alignment methods can be described as:

**Sequence (S)** Each landmark in the current run is aligned with the next unassigned landmark in the route description. If there is a mismatch between the number of landmarks in each run then there is no alignment for the excess landmarks.

**Distance (D)** The $x-y$ position odometry is transformed into a single dimension of distance travelled from the beginning of the run. Landmarks in the current run are aligned to the landmarks in the route description that are closest to them in distance travelled.

**Category and Distance (CD)** Each landmark in the current run is aligned to the landmark in the route description that is closest to the same distance travelled and is also a member of the same landmark category (given by the SOM). If there is no category match then no alignment is made for that landmark.

**Category and Sequence (CS)** Each landmark in the current run is aligned with the next unaligned landmark in the route description whose category it matches. If there is no category match then no alignment is made for that landmark.

**Category and Distance with Limited Range (CDL)** As in alignment CD, but matches are only allowed that are within $\nu(d)$ of the landmark at distance $d$ in the current run.

**Category and Sequence with Limited Range (CSL)** As in alignment CS, but matches are only allowed that are within $\nu(d)$ of the landmark at distance $d$ in the current run.

To evaluate the six methods quantitatively we looked at the each of the four scores taken over all the possible sets of test and training sets. The first question is whether there is any statistical difference at all between the various alignment methods. A Friedman ANOVA ($p = 0.05$, 32 blocks formed by the possible combinations of anticipatory models and choices of landmark detector settings, 54 trials/block) does reject the null hypothesis that all alignment methods are equally as good, on all four scores.

The next question is which alignment is best in each score. Table 1 answers that using a Tukey-Kramer multiple comparison of mean ranks ($p = 0.05$). The best looking alignments in terms of score look to be the D and S alignments. Closer inspection, however, soon shows them to be inferior in several ways. Although they do well on all other metrics, the lack of high category score is an indication that the landmarks are not actually being aligned to the same locations. This is obvious after a quick look at the algorithms, D and S aren't picky, they merely rely on the landmark detector being very consistent from run to run, and are likely to accumulate more error the longer the route. CDL has the highest scores of the remaining algorithms; importantly it has a high category score, indicating that it probably is aligning positions in the environment correctly.

A visual inspection of the example alignments in figure 1 illustrates the point, showing that the distance alignment makes many dubious landmark assignments because some landmark features are not detected on both runs through the environment. Looking at the same figure also demonstrates why the CDL align-

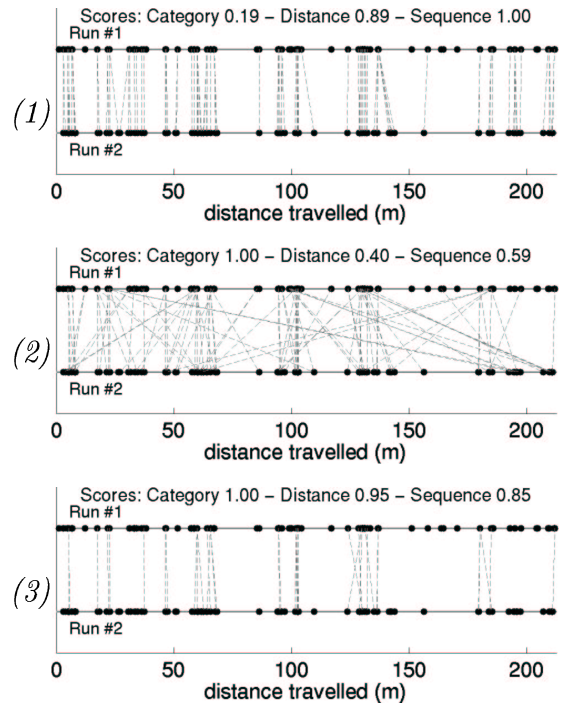


**Fig. 1.** Example alignments for a landmark detection system (standard single-layer network predictor, $n = 4$, using all sonars at landmark perceptions for SOM training) between two separate runs down the same route. The distance travelled along the route is the horizontal axis, landmarks are represented by black dots, and alignments by dashed lines between landmarks in the two runs. *(1)* Distance Alignment (D); *(2)* Category & Distance Alignment (CD); *(3)* Category & Distance Alignment with Limited Range (CDL)

ment does best – it finds fewer landmarks, but those that it finds are detected more consistently and at qualitatively better locations. This alignment method is therefore used to make all other comparisons in this paper.

It is easy to argue that the metrics proposed here are somewhat arbitrary and do not directly reflect the suitability of the algorithm to route navigation. Unfortunately it is not possible to externally check the correctness of the alignments since no accurate global position measurement is possible with these robots. However, these concerns are addressed by the experiments described in section 4.7, which show that CDL alignment is superiour to D alignment in real robot route navigation.

### 4.4 Effect of Parameters

This section investigates whether any particular choice of parameters for the anticipatory landmark detector are better than any others. Although the ANOVA

**Table 1.** Comparing alignment methods: mean scores, mean ranks, and statistically significant differences. See section 4.3 for a key of alignment abbreviations and an explanation of the scores. All statistically indistinguishable groupings, indicated by underlines, are due to a Tukey-Kramer comparison of rank means. ($p = 0.05$)

| **Category Score** | CD | CS | CDL | CSL | D | S |
|---|---|---|---|---|---|---|
| rank mean | 244 | 243 | 241 | 109 | 81 | 57 |
| score mean | 1.0 | 1.0 | 1.0 | .26 | .20 | .15 |
| **Distance Score** | CDL | D | CD | CSL | S | CS |
| rank mean | 269 | 264 | 170 | 93 | 91 | 88 |
| score mean | .87 | .83 | .51 | .26 | .26 | .25 |
| **Sequence Score** | S | D | CDL | CSL | CS | CD |
| rank mean | 260 | 260 | 162 | 157 | 74 | 61 |
| score mean | 1.0 | 1.0 | .82 | .81 | .70 | .68 |
| **% LM Aligned** | D | S | CD | CDL | CS | CSL |
| rank mean | 274 | 214 | 205 | 120 | 120 | 42 |
| score mean | 1.00 | 0.92 | 0.92 | 0.77 | 0.77 | 0.47 |

finds that there is a difference somewhere in our choice of parameters, the comparison of means is unable to pinpoint it. Instead, we qualitatively observe a few trends in the scores, and prefer to use the following settings: $n = 4$, using left-side only sonars for categorisation, trained on all locations in the environment.

We investigate three parameters in the landmark detector: detection threshold $n$, the choice of whether to use all sonars, or just the left-side sonars used to categorise landmarks, and how the categoriser is trained. The first setting is obviously important, but the other two can conceivably have a large effect on alignment methods that use category information. A Friedman ANOVA was used to investigate the treatments, which is shown in table 2. The treatments were taken over blocks representing the various anticipatory models, for the CDL alignment only. The ANOVA was performed separately for the distance scores, sequence scores, and percentage of landmarks aligned (it was not necessary to run the analysis on category score since this would be 1.0 in all CDL alignments). The ANOVAs all rejected the null hypothesis at a significance level of 0.05.

In spite of the ANOVA results, the Tukey-Kramer comparison at $p = 0.05$ (see table 3) could not tell which treatments were different from the others. Qualitatively, though, we can see that the left-side sonar models seem to do particularly well for the distance and sequence scores. Although the significance of this effect cannot be shown, we postulate that better scores could result from less 'noisy' categorisation; the wall-following side sonars would not be as affected by people walking through the hallways or minor changes of robot orientation. In general, using all perceptions outperforms using only the landmark ones to train the categoriser, probably allowing a better categorisation of the world. We

**Table 2.** The treatments investigated in section 4.4 to investigate the effects of landmark detection parameters on performance metrics.

| Treatment | Perceptions Used | Sonars Used | $n$ |
|---|---|---|---|
| I | Landmarks | Left | 3 |
| II | All | Left | 3 |
| III | Landmarks | All | 3 |
| IV | All | All | 3 |
| V | Landmarks | Left | 4 |
| VI | All | Left | 4 |
| VII | Landmarks | All | 4 |
| VIII | All | All | 4 |

also notice that the percentage of landmarks aligned does seem to decrease with increasing detection threshold. Therefore we generally prefer treatment VI for subsequent experiments.

**Table 3.** Comparing the effects of SOM inputs and landmark detection threshold: mean scores and mean ranks. No statistically significant differences could be found between settings using a Tukey-Kramer comparison of rank means at significance level 0.05. See section 4.4 for a key of the group labels and an explanation of the parameters that were varied

| **Distance Score** | II | I | VI | V | IV | VIII | III | VII |
|---|---|---|---|---|---|---|---|---|
| rank mean | 266 | 265 | 255 | 249 | 181 | 174 | 173 | 170 |
| score mean | 0.91 | 0.91 | 0.91 | 0.89 | 0.88 | 0.86 | 0.88 | 0.85 |

| **Sequence Score** | VI | II | VIII | V | IV | I | VII | III |
|---|---|---|---|---|---|---|---|---|
| rank mean | 256 | 243 | 227 | 227 | 221 | 208 | 186 | 163 |
| score mean | 0.79 | 0.81 | 0.77 | 0.76 | 0.78 | 0.77 | 0.74 | 0.74 |

| **% LM Aligned** | II | VI | I | V | III | IV | VII | VIII |
|---|---|---|---|---|---|---|---|---|
| rank mean | 302 | 290 | 277 | 255 | 165 | 162 | 145 | 135 |
| score mean | 0.65 | 0.63 | 0.61 | 0.59 | 0.46 | 0.46 | 0.43 | 0.41 |

### 4.5 Selecting an Anticipatory Model

In this section we investigate how varying the complexity of the anticipation model affects alignment. We find that although there are differences between the various networks, there is no network that is best in all scores. We also find that, in general, the naïve predictor not only scores poorly, but also produces landmarks that are not very useful when looked at qualitatively. In general, we prefer the standard anticipatory model due to it's combination of simplicity and good performance.

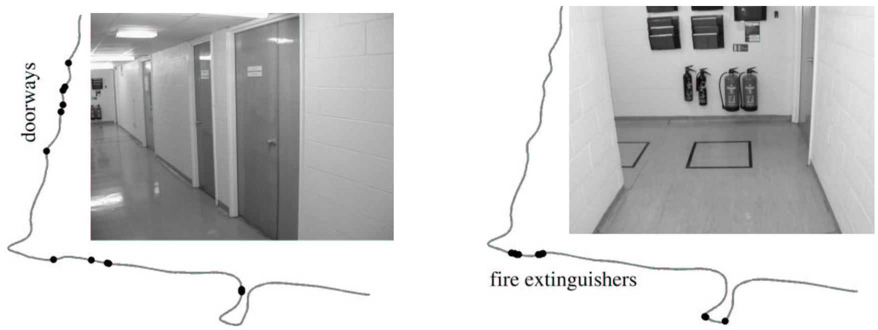


**Fig. 2.** An example of the differences in landmark selection between the *(left)* standard anticipatory model and the *(right)* GWR non-anticipatory model, $a_T = 0.85$. All methods trained using the same parameters on the same data and tested on the same data (different from training set). Landmarks found are shown by black dots on the line of the route path. They correspond to physical features on the route: an alcove, fire extinguishers on the wall *(inset right)*, and a series of closed doorways*(inset left)*

Separate Kruskal-Wallis ANOVAs were run for each score, with the treatments being of the four anticipatory models described in section 2.1 plus the non-anticipatory GWR algorithm described in section 3. Discussion of the GWR results and the comparison of the two types of landmark detector is left to section 4.6. Blocking is no longer necessary since all tests are performed using the same alignment method (CDL, see section 4.3) and parameter settings (VI, see section 4.4). All ANOVAs rejected the null hypothesis at $p = 0.05$. A Tukey-Kramer multiple comparison of mean ranks at the same significance level can be seen in table 4.

Although all three networks were indistinguishable in terms of distance score, the naïve predictor had far worse distance performance. In sequence score, the naïve predictor comes out on top, with the networks behind. Finally, in terms of percentage of landmarks aligned, the naïve approach performs poorly compared to the other predictors. Because the naïve approach eliminates the neural network it lacks context both in terms of sensory prediction and also because the noise model of the Kalman filter changes, losing the prediction term. The combined effect leaves the naïve predictor finding landmarks only where very large single time-step changes in sensor values occur. Since such places are rare, it turns out that the naïve predictor finds about an order of magnitude fewer landmarks than the other anticipatory models. Additionally, the landmarks are very spread out; there are very few cases of such landmarks occurring in successive perceptions, and therefore little chance for the landmarks to be aligned out of sequence. Overall, the nature of the naïve landmark predictor is fairly unsuitable; the landmarks that it finds very large rapid changes of sensor value, such as cracks in the wall. Such landmarks are not sensed reliably in every run through the environment.

**Table 4.** Comparing the effects of different landmark selection models: mean scores, mean ranks, and statistically significant differences. Three versions of single-layer sensory anticipation network *(Standard, Recurrent, Lagged)* are compared along with a naïve anticipator that always guesses the current perception and a non-anticipatory landmark selector *(GWR)*. See section 4.5 for an explanation of the comparisons. All statistically indistinguishable groupings, indicated by underlines, are due to a Tukey-Kramer comparison of rank means ($p = 0.05$).

| **Distance Score** | Recurrent | Standard | Lagged | GWR | Naïve |
|---|---|---|---|---|---|
| rank mean | 983 | 966 | 952 | 850 | 670 |
| score mean | 0.89 | 0.90 | 0.88 | 0.87 | 0.82 |

| **Sequence Score** | Naïve | GWR | Lagged | Standard | Recurrent |
|---|---|---|---|---|---|
| rank mean | 1378 | 1351 | 775 | 730 | 624 |
| score mean | 0.93 | 0.94 | 0.77 | 0.79 | 0.75 |

| % **LM Aligned** | Recurrent | Standard | Lagged | Naïve | GWR |
|---|---|---|---|---|---|
| rank mean | 1123 | 1096 | 951 | 479 | 236 |
| score mean | 0.56 | 0.55 | 0.49 | 0.29 | 0.18 |

### 4.6 An Alternative to Anticipation

In this section we compare a non-anticipatory, novelty filter approach (the GWR network described in section 3) to the previous landmark detectors. The novelty filter is shown to have a trade-off between generalisation and landmark detection. When the insertion threshold $a_T$ is set high enough to ensure that at least a few landmarks are found, it tends to have too fine-grained a categorisation of the environment to reliably align itself.

The GWR network was investigated for several levels of node insertion threshold, between $a_T = 0.6$ and $a_T = 0.85$. The insertion threshold controls how similar two inputs have to be before the same node in the network will be used to represent them. A value close to one will require that each node matches the input very accurately, so that many nodes will be generated; a smaller value will generate fewer nodes to represent the same inputs. Informal experiments showed that low values of the insertion threshold do not discriminate between the sonar values seen during typical exploration by the robot, so that occasionally no landmarks at all are selected during a test run of over 200 m. We therefore used an insertion threshold of 0.85 to compare the performance of the GWR network with the various models described in the previous section.

The null hypothesis, that all anticipatory models and the GWR network perform the same at landmark detection, was rejected in section 4.5 and a comparison of score rank means can be seen in table 4. It can be seen that for distance and sequence scores the GWR performed comparably to the naïve predictor, but found very different landmarks. It is interesting to note that the naïve predictor finds very few landmarks, an average of 20, yet it is able to align them more often (% landmarks aligned) than the GWR network, which finds an average of

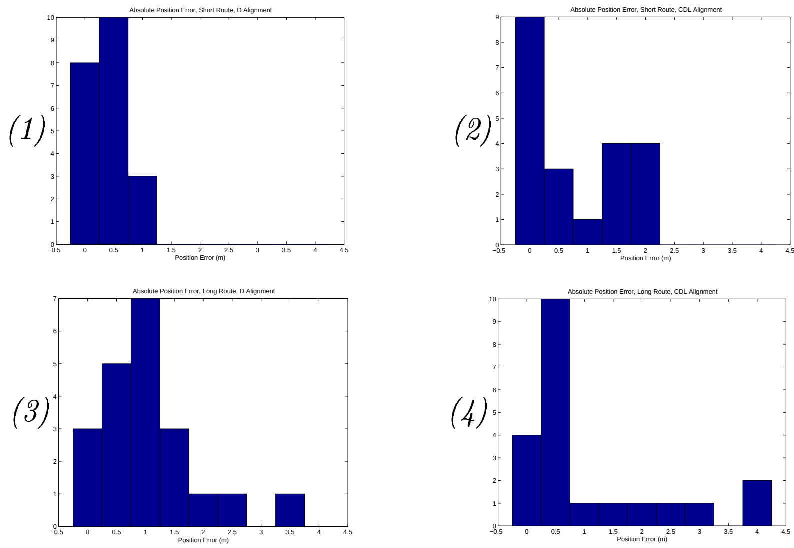


**Fig. 3.** Histograms of the absolute displacement error for each of the 21 following runs, for *(1)* D alignment over the short trip, *(2)* CDL alignment over the short trip, *(3)* D alignment over the long trip, and *(4)* CDL alignment over the long trip. A run with a displacement error of more than about 0.6 m would be very unlikely to make it into the doorway successfully

68 landmarks at $a_T = 0.85$. This may happen because the typical GWR size is 114 nodes, compared to just 20 in the case of the SOM. The GWR network finds a more fine-grained categorisation of the world, and therefore is less likely to find an exact category match for alignment. This hampers its performance in localisation using the CDL comparison from section 4.3. However, the best intuition as to what is going on can be had by examining figure 2. GWR only finds landmarks that have never been seen previously, like the fire-extinguishers on the wall. Naïve prediction (not shown in figure) finds only a single landmark at the entrance to the alcove, a place where there is a very large sensor value change between successive perceptions. But the full anticipatory model finds landmarks wherever perceptions change in relation to the recent history of perception, therefore landmarks are close enough together to be useful in correcting odometry drift.

### 4.7 Landmark Detection In Use: A Door-finding Experiment

In this section we use the anticipatory landmark detector in a goal-oriented navigation task. We compare success rates at the task using D and CDL alignment methods for localisation, and show that CDL alignment performs better than D alignment, as predicted by section 4.3. We also note that the CDL alignment shows no obvious degredation of performance when the distance travelled is more than doubled.

The task is to start from either a close (about 22 m straight travel) or distant (about 68 m travel with two right-angle turns) starting point, navigating along a corridor with no possible branches along the route. At the open door, which is approximately 1.8 m wide, the robot must execute a 90 degree turn and attempt to find its way through the door using an obstacle avoidance behaviour. The robot performs a baseline run first, during which is was told to turn at the door by the experimenter. The robot then records the route description as the series of landmark categories perceived along the route with the relative distances between the landmarks and the action (such as 'wall-follow' or 'turn left') to perform at each landmark. The robot is then returned to the starting location and attempts to duplicate the route by using either the CDL or D alignment methods for localisation. In all, three baseline runs were performed in the short hallway, and three in the long. Each baseline run was followed by seven attempts at following the route directions to the door again. The number of successes in navigating through the door was recorded and the difference between the positions at which the robot stopped when going through the door were compared to the original training run.

A Kruskal-Wallis ANOVA ($p = 0.05$) rejected the null hypothesis that the four possible treatments (CDL long, CDL short, D long, D short) were equivalent in success rate. A Tukey-Kramer comparison (see table 5) finds a significant difference between the long distance CDL and the long distance D alignments, which validates our earlier results that using category and odometry measurements aid robot localisation. In fact, this can be seen in the histogram of displacement errors in figure 3. The CDL alignment has a sharper peak near zero displacement error than the D alignment for both lengths of route. The amplitude of this peak is also unchanged with distance, unlike the D alignment. In both alignments though, the total spread of the errors is roughly proportional to the distance travelled, which is not surprising for the D alignment given our model of odometric error on the robot. In fact, one can fit a gamma distribution to the D alignment errors using maximum likelihood estimators and find that the 5% odometric drift model corresponds to roughly five standard deviations normalised against distance travelled. But the CDL alignment also displays this distance proportional error spread because if it misses making the category match it will continue until the odometry error model says the robot has gone past any reasonable hope of finding the doorway before stopping.

**Table 5.** A Tukey-Kramer comparison ($p = 0.05$) of rank means for the four cases of the door-finding experiment. Statistically insignificant difference between means are grouped together by the underlines. The CDL alignment method is found to function significantly better than the D alignment when the travel distance is longer.

| **Success rate** | CDL long | CDL short | D short | D long |
|---|---|---|---|---|
| rank mean | 50 | 46 | 42 | 32 |
| score mean | 0.66 | 0.57 | 0.48 | 0.24 |

## 5 Discussion

Given how much computational machinery is at work in the anticipatory landmark selector, it is important to ask how much of it is actually necessary. There are three major components: a sensory anticipation model, a Kalman filter operating on the error of the sensory anticipations to detect landmarks, and a mechanism for producing categorisations of the resulting landmarks.

Firstly, the anticipation model needs to be of a useful complexity. Both here and in previous work [14, 6] it has been demonstrated that a single-layer network is sufficient for the sensor prediction. In fact, it is preferable to multi-layer networks, because the more complex model may learn to predict so well that there are never any landmarks [14]. In this paper, as in [14], it has been shown that variations in the network produce little difference in terms of score, but the network with lagged inputs does seem to learn to predict the environment better than the other two, producing fewer, better quality landmarks. In comparison, the naïve approach of always predicting the current perception is insufficient. It can only detect landmarks where perceptions change by a large amount very quickly; such landmarks are often not detected on every trip through an environment. Learning any kind of model of the environment allows for a better selection of landmarks.

The Kalman filter is necessary because peak detection on the model output produces an unwieldy number of poor quality landmarks [14]. The filter provides a way to remove error due to both sensor measurement variation and robot orientation. But could the Kalman filter operate directly on the sonar inputs, removing the need for the prediction network? The problem with this approach is that the sensor measurement variance must be modelled in the filter. In the anticipatory model the raw variance in the sensor measurements – an experimentally measurable quantity – is propagated through the non-linearity of the predictive network and the sum-of-squared errors to form a term of SSE variance due to sonar variation [6]. This value is always non-zero due to the summation of variance across all of the sonars. But a problem occurs if the filter operates directly on the sonars. The variance of the sonar perpendicular to the wall is almost zero, since sonars have very little measurement error for hard surfaces that have small angles of incidence to the beam. In the Kalman update equations, the gain will tend to unity and $\mathrm{Var}[\widehat{E}]$ will be close to zero, so that the filter would see every perception as a landmark unless some clever voting scheme were introduced that removed the effects of the sonar that is perpendicular to the wall.

Another good reason for using the filter is that both the recursive nature of the filter and the time dependency of the noise model (the sum-of-squared errors have terms at time $t$ and time $t-1$) provide additional context to the landmark detection system. The combination of context in the filter and context in the anticipatory sensor model dictate that landmarks selected with this system are distinct in comparison to recent landmarks, something that we believe to be especially important when using landmarks to localise. Assuming that the robot is in approximately the right area – something possible even with pure odometry over medium distances – accurate localisation is aided by individual landmarks being distinctive.

We used a SOM to categorise landmarks found by the anticipatory landmark detector because it is a well-known self-organising algorithm, but many other algorithms could be substituted to produce landmark categorisations. We could not find a definitive reason to prefer one choice of training perceptions or sensor inputs over another, in spite of the fact that using categories in the method of aligning and choosing landmarks radically improves their performance. It seems that the use of categories in alignment is more important than how those categories are created, although this has not been fully investigated. It does seem likely, however, that the number of categories has an effect on landmark alignment. A large number of categories would result in a very fine-grained categorisation of the environment, something equivalent to 'a T-junction with a scratch on the wall' rather than the more useful general category of 'T-junction'. Without a method to cluster similar categories the assumption that a landmark must exactly match in category will break down.

In general, the non-anticipatory GWR novelty filter performed on a similar level to the naïve anticipation model, but picked very different landmarks. One of the problems with this method of landmark selection is that when the insertion threshold is raised high enough to select a useful number of landmarks, the categorisation of the environment is very fine-grained, and therefore only a small percentage of the overall number of landmarks is actually aligned. As this method is not based on anticipation, any changes that the robot experiences that were not seen during training (for example, the robot travels closer to the wall that it did in training) will be detected as landmarks. An anticipatory system can deal better with these changes. Furthermore, this perceptual novelty system finds very different landmarks because it highlights locations that are different from the entire history of perceptions, rather than locations that differ given the recent context. This process potentially excludes a whole class of landmarks that could be useful: landmarks that are distinct compared to their surroundings even though they might be something that is familiar to the robot.

## 6 Conclusions

This paper has demonstrated that sensory anticipations, even very simple ones, can be valuable in selecting landmarks for a route-following system. It has presented both an anticipatory landmark detector that selects landmarks based on novelty compared to recent perceptions, and a non-anticipatory landmark detector that selects landmarks based on novelty compared to all previous perceptions. These two methods produce qualitatively and quantitatively different landmarks, resulting in different levels of landmark alignment when a robot tries to align it's current perceptions of a route with landmarks that it has seen when following the route previously.

In general, we would prefer to use the anticipatory landmark detector for a route-following task. It produces higher scores in our landmark alignments, perhaps because the perceptual novelty algorithm produces a finer-grained categorisation of the environment. The anticipatory method also has the theoretical advantage of being based on novelty in the local context, which can make the alignment of landmarks easier if the robot is looking for a match when it already knows that it is in the correct general area. We also find that the complexity of the predictive model must be greater than naïvely predicting the current perception, but we also know from previous experiments that a very complex model that can predict the environment very well will never have enough error to produce a landmark. The simple single-layer artificial neural networks presented here seem to offer a good compromise.

The paper also shows that combining category, distance, and odometric error information produces better alignments between landmarks in different runs – and therefore better route following – than merely following metric directions. This does not mean that more complex methods of localisation (e.g., multiple-hypothesis tracking or belief revision based on temporal evidence) are not necessary, but we do find that these simple methods can allow reasonable navigation success on routes on the order of 50 m.

## Acknowledgements

Part of this work was supported by a Canadian & U.S. Citizens' Scholarship and an ATLAS grant from the Department of Computer Science at the University of Manchester. We would like to thank Tom Duckett, Ulrich Nehmzow, and David Gelder, who were involved in earlier publications on this work. We would also like to thank Susannah Lydon, Richard Craggs, and the anonymous reviewers for commenting on earlier versions of this paper.

## References

1. A. Baraldi and E. Alpaydin. Constructive feedforward ART clustering networks - part II. *IEEE Transactions on Neural Networks*, 13(3):662–677, 2002.
2. A. Baraldi and P. Blonda. A survey of fuzzy clustering algorithms for pattern recognition: Part II. *IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics*, 29(6):786–801, December 1999.
3. Eric Bourque and Gregory Dudek. On-line construction of iconic maps. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'00)*, pages 2310–2315, 2000.
4. Martin V. Butz, Olivier Sigaud, and Pierre Gerard. Internal models and anticipations in adaptive learning systems. In *Lecture Notes in Artificial Intelligence*. Springer, this volume.
5. Tom Duckett and Ulrich Nehmzow. Mobile robot self-localisation using occupancy histograms and a mixture of gaussian location hypotheses. *Robotics and Autonomous Systems*, 34(2–3):119–130, 2001.
6. Jason Fleischer and Stephen Marsland. Learning to autnomously select landmarks for navigation and communication. In *Proceedings of the Seventh International Conference on Simulation of Adaptive Behavior*, pages 151–160. MIT Press, 2002.

7. Myles Hollander and Douglas Wolfe. *Nonparametric Statistical Methods.* Wiley, New York, 1973.
8. Dan-Anders Jirenhed, Germund Hesslow, and Tom Ziemke. Exploring internal simulation of perception in mobile robots. In *Proceedings of the Fourth European Workshop on Advanced Mobile Robots*, pages 107–113, 2001.
9. R.E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82:34 – 45, March 1960.
10. Teuvo Kohonen. Self-organised formation of topologically correct feature maps. *Biological Cybernetics*, 43:59–69, 1982.
11. David Kortenkamp and Terry Weymouth. Topological mapping for mobile robots using a combination of sonar and vision sensing. In *Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI'94)*, pages 979–984, Seattle, Washington, 1994.
12. George Maistros, Yuval Marom, and Gillian Hayes. Perception-action coupling via imitation and attention. In *Proceedings of the AAAI Fall Symposium on Anchoring Symbols to Sensor Data in Single and Multiple Robot Systems*, pages 52–59, 2001.
13. Stephen Marsland. Novelty detection in learning systems. *Neural Computing Surveys*, 3:157–195, 2003.
14. Stephen Marsland, Ulrich Nehmzow, and Tom Duckett. Learning to select distinctive landmarks for mobile robot navigation. *Robotics and Autonomous Systems*, 37:241 – 260, 2001.
15. Stephen Marsland, Ulrich Nehmzow, and Jonathan Shapiro. A self-organising neural network that grows when required. *Neural Networks*, 15(8-9):1041–1058, 2002.
16. Volker Stephan and Horst-Michael Gross. Neural anticipative architecture for expectation driven perception. In *Proceedings of 2001 IEEE International Conference On Systems, Man & Cybernetics*, pages 2275–2280, 2001.
17. Sebastian Thrun. Bayesian landmark learning for mobile robot localisation. *Machine Learning*, 33(1):41 – 76, 1998.
18. Juha Vesanto, Johan Himberg, Esa Alhoniemi, and Juha Parhankangas. SOM Toolbox for Matlab 5. Report A57, Helsinki University of Technology, Neural Networks Research Centre, Espoo, Finland, April 2000.
19. Daniel M. Wolpert and Zoubin Ghahramani. Computational priniciples of movement neuroscience. *Nature Neuroscience Supplement*, 3:1212–1217, 2000.
20. Brian Yamauchi and Randall Beer. Spatial learning for navigation in dynamic environments. *IEEE Transactions on Systems, Man and Cybernetics Section B*, 26(3):496 – 505, 1996.