Context-based sound event recognition
Niessen, Maria Elisabeth

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2010

Link to publication in University of Groningen/UMCG research database

Citation for published version (APA):

Copyright
Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license. More information can be found on the University of Groningen website: https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment.

Take-down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): http://www.rug.nl/research/portal. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.
Artificial agents that operate in a real-world environment have to process an abundance of information, which may be ambiguous or noisy. We present a method grounded in cognitive research that keeps track of sensory information, and interprets it with knowledge of the context. We test this model on visual information from the real-world environment of a mobile robot in order to improve its self-localization. We use a topological map to represent the environment, which is an abstract representation of distinct places and the connections between them. Expectancies of the place of the robot on the map are combined with evidence from observations to reach the best prediction of the next place of the robot. These expectancies make a place prediction more robust to ambiguous and noisy observations. Results of the model operating on data gathered by a mobile robot confirm that context evaluation improves localization compared to a signal-driven model.
6.1 INTRODUCTION

Artificial agents that operate in a real-world environment are confronted with additional challenges compared to agents that operate in a controlled or simulated environment. They have to process an abundance of information, of which not everything is necessarily relevant for their goal. Moreover, sensory information may be ambiguous or noisy. To make sense of its environment, an agent needs to identify and structure the sensory information it gathers. We developed a method grounded in cognitive research that keeps track of sensory information, and interprets it with knowledge of the context (chapter 4).

Applications of cognitive research, such as handwriting recognition (Côté et al., 1998) and information retrieval (Crestani, 1997; Van Maanen et al., 2010), often employ a spreading activation semantic network to recognize a particular item or retrieve specific information. Spreading activation networks are based on a model of human memory (Quillian, 1968). They are realized as connected nodes that represent pieces of information or concepts, and the vertices represent the prior probabilities that the nodes are encountered together. Spreading activation networks are typically static, because the data in these application domains can be accessed completely and simultaneously. In contrast, for agents operating in a dynamic environment the available information continuously changes.

To deal with continuous data, we apply a context model that manages a dynamic network. This dynamic network is similar to a spreading activation network, but instead of being static, it is updated when new data are encountered. The model continuously updates its current state, based on sensory input and knowledge of the context. The context model has been applied to recognize sound events (see chapter 5), but is designed to manage any type of sensory input. We will show in this chapter that it can also be applied to visual information from the real-world environment of a mobile robot.

A basic task for an autonomous mobile robot is to build a map of its environment for self-localization. For this reason, simultaneous localization and mapping (SLAM) has received considerable attention in the last decade. Most SLAM approaches use range or vision sensors to construct a detailed metric map of the environment (Thrun et al., 2005). These maps contain the Cartesian coordinates of many structural features present in the environment. Other approaches build
topological maps of the environment (Vasudevan et al., 2007). Instead of representing the environment in detail, it is represented more abstractly in topological maps, as distinct places and the connections between them. The advantage of such an abstract representation is that it is less susceptible to noise, and ambiguous observations and situations. Moreover, it results in a computationally less demanding system.

In topological mapping, a rough estimate of the location of the robot can help to form an expectancy of the path of the robot. This expectancy can be combined with evidence from observations to form a hypothesis of the place of the robot. Furthermore, an expectancy of the place of the robot can resolve ambiguous observations. In this way, the place in a topological map where an observation is made can be considered as the topological context of that observation. When the robot is moving and making observations, an evaluation of the context can improve its localization. The evaluation of the context entails that the recent history of visited places is used to predict the place that follows. Furthermore, using knowledge of the topological context makes localization more robust to noise in the observations.

In the next section we describe the design of the model, and how it processes observations made by a mobile robot. In section 6.4 we present the results of two experiments that are described in section 6.3. The first experiment demonstrates that the model is more robust to noise when the topological context is used. The second experiment shows that predictions in real data with many ambiguous observations and noise are also better with context evaluation than without. We end with a discussion on the performance of the model and give an outlook on future work.

### 6.2 Methods

The model we present processes visual input of a moving robot. These visual observations, which are explained in section 6.2.1, provide evidence about the place of the robot. However, ambiguous or noisy observations can lead to erroneous place predictions. To improve these predictions, contextual information about the environment is learned in a supervised training phase and stored in a static knowledge network. In the operation phase this knowledge is used in a dynamic network,
which computes expectancies of the place of the robot.

The knowledge about the environment, in the form of nodes in the knowledge network and the strength (weight $w$) of the connections between them, is computed in the training phase. We refer to this knowledge as long-term memory, since it reflects stable knowledge. Therefore, it is stored as a static network, which is constructed from learning relations in the training data. This knowledge network is similar to semantic networks used in information retrieval. In section 6.2.2 we describe in more detail how the knowledge network is created.

In contrast to the knowledge network, the dynamic network reflects short-term memory. Information represented by nodes in this network is added and forgotten more quickly, since the nodes pertain only to the current state of the robot. Nodes in the dynamic network are called hypotheses, because they represent possible explanations for input data. The dynamic network has three levels that all represent a different type of information: hypotheses of observations, landmarks, and places in the environment. Figure 6.1 shows an example of a dynamic network at one moment, namely when observation 2 has been made. The network configuration represents the knowledge of the environment at that moment. This knowledge consists of two observations, their connections to landmarks hypotheses, and the connections of the landmarks to hypotheses of places in the environment. In section 6.2.3 we explain the construction of the dynamic network, and how topological context is used to compute expectancies of the place of the robot.

6.2.1 Observations

The robot (a Pioneer 2 DX mobile) uses a video camera to navigate in its environment. Visual interest points are detected in the camera images, which serve as landmarks to represent the environment. The interest points are detected and described using the scale-invariant feature transform (SIFT, Lowe, 2004). The SIFT algorithm detects points that stand out from their surroundings. These points are described using histograms of gradients. A drawback of SIFT is that it results in a large number of interest points, many of which are not re-detected in subsequent images. Therefore, we use a visual buffer to test the stability of the interest points over a number of successive images (Kootstra et al., 2009). Only interest points that are stable enough are used as landmark observations. The descriptor of an
Figure 6.1: Example network configuration at one instant, of two observations that are matched to three landmarks, each in turn connected to a place.

observation is then compared to that of previously observed landmarks. Based on the descriptor distances, the observation is matched with one or more landmarks or labeled as a new landmark.

The data set used in one of the two experiments (section 6.3.2) was collected by the robot while it drove a closed loop of eight by ten meters in an office-like environment. The data was logged by the robot while driving four laps. The map of the loop was manually divided into nine places, as depicted in Figure 6.2. Half of the data set, that is, the observations made in the first two laps, is used to determine which landmarks are observed in which place. The other half is used to test the model. Because of the variability of the images in different laps, the robot might have observed landmarks in the last two laps that are not present in the training data.
Figure 6.2: Environment where the robot drove four laps. The size of the loop is 8 by 10 meters, divided into nine places. The gray area consists of objects the robot cannot drive through.

6.2.2 Knowledge network

Three classes of information are stored in a knowledge network: the descriptors of the landmarks, the relations between the landmarks and the places in the environment, and the transitions between the places. This knowledge network represents the context, which is slowly changing or invariant. Therefore, it is referred to as the long-term memory of the model.

The connection strengths between landmarks and places in the training data are calculated according to a term-weighting approach used in automatic document retrieval (Salton and Buckley, 1988). In this method the importance of a term (word or phrase) in a document is determined by multiplying its frequency in the document (term frequency) with its general frequency in other documents (inverse document frequency). Hence, the term is important for a document if it occurs often in that document and infrequently in other documents. Since the connection
strength (weight $w$) between a landmark and a place should reflect the specificity of the landmark to that place, we adopt the term-weighting approach. The landmarks can be treated as terms, and the places as documents. Accordingly, the weight of the connection between landmark $l$ and place $r$ is:

$$w_{r,l} = w_{l,r} = \frac{\log_{10} N - \log_{10} n}{\log_{10} N}$$  \hspace{1cm} (6.1)

where $N$ is the total number of places, $n$ is the number of places in which landmark $l$ is observed, and the normalized term frequency is given by:

$$tf = \frac{f_{l,r}}{\sqrt{f_l}}$$  \hspace{1cm} (6.2)

where $f_{l,r}$ is the observation frequency of $l$ in $r$, and $f_l$ is the total observation frequency of $l$.

The connections between observations and landmarks are not stored in the knowledge network, because all observations are unique. Therefore, the weights of these connections are computed at the moment when an observation is made, both in the training and the operation phase.\(^1\) The connection strength between an observation and a landmark should represent the likelihood of a correct matching between their descriptors. If these descriptors are far apart, the observation and landmark are less likely to have been matched correctly. Therefore, the weight of a connection between an observation and a landmark is inversely related to the distance between their descriptors:

$$w_{l,o} = w_{o,l} = 1 - \frac{d}{\theta_d}$$  \hspace{1cm} (6.3)

where $d$ is the distance between the descriptor of observation $o$ and landmark $l$, and $\theta_d$ is the maximum distance at which an observation is still matched to a known landmark.

The transition probability that the robot moves from one place to another is calculated by normalizing the number of times the robot moves from one place

\(^1\) Observed information is not necessarily always unique. In other domains or applications areas it could be useful to store observations in the knowledge network. However, in the presented application it would be useless to do so.
to another in the training data (the first two laps). As can be seen in Figure 6.2, the robot can move within place \( i \) or move from place \( i \) to place \( i \pm 1 \). Since the robot is driving the loop in one direction, the transition probabilities to all places other than \( i \) and \( i + 1 \) are generally zero. However, there are a few exceptions when no observations are made in a place in one of the laps, and thus the probability to move to \( i + 2 \) is greater than zero. The complete matrix of probabilities serves as the topological context that helps to compute an expectancy about the next location of the robot.

To summarize, the knowledge network consists of the matrix with the a priori transition probabilities between all places. Furthermore, it stores the labels of all landmarks that are observed in the training data, along with their connections to the places in which they are observed.

### 6.2.3 Dynamic network of hypotheses

Once the knowledge network is fully trained after the learning phase, it is used in the operation phase, together with evidence from observations, to predict the place of the robot. The algorithm for the construction and updating of a dynamic network is summarized in Table 6.1 (see chapter 4 for a detailed description). Every level in the network consists of hypotheses of a single type of representation (see Figure 6.1). The landmark observations are the lowest level of the dynamic network. As described in section 6.2.1, observations are matched to one or more previously observed landmarks, or labeled as a new landmark, which are at the middle level. The highest level in the network holds hypotheses of places in the environment.

Each node in the network represents a hypothesis of one of the three different types of representation. When an observation is made, a hypothesis is added to the dynamic network (step 1). Next, its matched landmarks (that are stored in the knowledge network\(^1\)) are initiated as hypotheses, and they are connected to the observation hypothesis (step 2). Subsequently, these landmark hypotheses retrieve their place connections from the knowledge network. These places are also initiated as hypotheses (step 3) and connected to the landmark hypotheses that initiated

---

\(^1\) The current version of the model only processes known landmarks in the operation phase. The possibility to add new landmarks will be discussed in section 6.5.
Table 6.1: Algorithm for updating the dynamic network configuration at times when observations are made by the robot.

For all observations at time $t$:

1. Add observations $O(t) = \{o_{t,i}\}$ to the network
2. For each $o_{t,i} \in O(t)$ add matched landmark hypotheses $\{l_{t,j}\} \in L(t)$ and connect them with strength $w_{i,j}$
3. For each new landmark $l_{t,j} \in L(t)$ add appropriate places $r_k$ not yet present in the set of active places $R$
4. Connect each new landmark $l_{t,j}$ with the appropriate place $r_k$ with strength $w_{j,k}$
5. Spread signal-driven activation
6. Spread context-based activation
7. Evaluate activation values

them (step 4). Every time new observations are made, the network is updated and the dynamics change.

Activation spreading

After the connections in the network are updated, the activation of the observation hypothesis spreads through the network (see section 4.3). The input activation first spreads upward to the place hypotheses at the highest level in the network, and is called signal-driven spreading (step 5). Subsequently, the activations of the place hypotheses spread downward to other connected hypotheses, for example landmarks in the same place that are observed previously. We call this context-based spreading (step 6). As a consequence of context-based spreading, a landmark hypothesis of a particular observation can be reinforced by later observations. For example, in Figure 6.1 the first observation is matched to landmarks 1 and 2, where landmark 1 lies in place A and landmark 2 in place B. Another landmark observation made in place B will increase the support for the hypothesis that the first observation was of landmark 2, and not of landmark 1.

Activation evaluation

After the activation has spread through the network, the activation value of each hypothesis is evaluated (step 7). The activation evaluation is different for different
types of hypotheses. The activations of the hypotheses that are not at the highest level in the network are normalized (equation 4.3). However, the place hypotheses have an expected activation value, like sound events in fixed sequences (section 5.2.2), because the order in which the robot drives through the environment is not random. Therefore, the activations of the place hypotheses at the highest level are a weighting of evidence from the input and an expected value.

The expected activation of place hypotheses represents the expectancy to be at a place given the context. It is calculated using the information about the place transitions in the environment (Figure 6.2). The expected activation of place \( i \) is the sum of all possible options to drive to place \( i \):

\[
\hat{A}_i(t) = \sum_j f_j(\Delta t)A_j(t - \Delta t)P(j \rightarrow i)P(j) \quad \text{for } i, j \in R, \tag{6.4}
\]

where \( A_j(t - \Delta t) \) is the previous activation of place hypothesis \( j \), multiplied with a decay \( f_j(\Delta t) \), \( P(j \rightarrow i) \) is the transition probability from place \( j \) to place \( i \), including \( j = i \), the probability to stay in the same place. Finally, \( P(j) \) is the a priori probability to be in place \( j \), and \( R \) is the subset of hypotheses that represent places.

The a priori transition probabilities from equation 6.4 are retrieved from the knowledge network. The probabilities are adjusted in the dynamic network of hypotheses, because the probability that the robot leaves a place increases as it is longer in that place. More specifically, the probability of staying in the same place decreases as a function of the age \( T_i \) (how long it is active) of the place hypothesis: \( P(i \rightarrow i)(T_i) = P(i \rightarrow i)^{T_i} \). The probabilities to move to other places are increased proportionally to their a priori connection strength. For example, suppose the initial transition probability between place A and place B is 0.2, and the probability of staying in place A is 0.8. After the robot has observed landmarks in place A at four subsequent times, \( P(A \rightarrow A) = 0.8^4 = 0.4 \) and \( P(A \rightarrow B) = 0.6 \). When the robot returns to the same place, the probabilities are re-initialized to the probabilities in the knowledge network.

The expected activation is combined with evidence from the current input to compute the activation evaluation of the place hypotheses:

\[
A_i(t) = \hat{A}_i(t) + K \left( \frac{n_i(t)}{\max n(t)} - \hat{A}_i(t) \right) \quad \text{if } i \in R, \tag{6.5}
\]
where $\hat{A}_i(t)$ is the expected activation according to equation 6.4, $n_i(t)$ is the input activation of $i$ as calculated in equation 4.1, $n(t)$ is a list with the input activations of all active place hypotheses, and $K$ is the gain factor. The gain factor is dependent on the noise in the observations. If the observations are very reliable, its value should be high. However, the current data set is relatively noisy. Therefore, the gain factor is set to 0.25, which entails that the model responds relatively slowly to new observations, and is guided more by expectancies.

The final activation values of all active place hypotheses are compared, and the one with the highest activation is the current best hypothesis of the place of the robot. Hence, the sequence of best hypotheses at each update gives the estimation of the model of the path of the robot.

6.3 EXPERIMENTS

To illustrate the benefit of context evaluation in robot localization, we show the place predictions of two models. In the first model the predictions are based on instant observations alone, which implies that only information from the knowledge network is used. Accordingly, context-based spreading is not applied, because the signal-driven model does not remember previous predictions. In other words, hypotheses of the place of the robot are deactivated after the signal-driven activation spreading. In the second model, the context-based model, the place prediction is based on a combination of instant observations and expectancies, which are computed through context evaluation, as discussed in section 6.2.3 ($\tau = 67$ in equation 4.2, activation threshold $\theta_A = 0.05$, and $K = 0.25$ in equation 6.5).

We discuss the results of both models running on two types of data. In the section 6.3.1 we present an experiment with simulated data, which can be controlled in their complexity. The simulated data are a simplification of the real data described in section 6.2.1. The experiment with the real data is discussed in section 6.3.2.

6.3.1 Simulated data

We generated a data set to measure the performance of the model on data with different levels of noise. The noise simulates observations that are so similar that they are matched to the same landmark, although the observations are made at
distinct places. These types of ambiguous observations occur often in the real data due to reoccurring objects and structures in office-like environments. At every time step one observation is simulated, which is matched to one landmark. The distance between the descriptor of the observation and the landmark is set to the same value for all observations. In the first lap all 240 landmarks are observed and connected uniformly to one of eight places. No noise was applied in the training part of the data, so there are no ambiguous landmarks in the a priori knowledge network. In the test data we applied a varying amount of noise on the landmarks. When no noise is applied to the data, the test set is identical to the training set. As the amount of noise increases, the place at which a landmark is observed becomes more random, until it is completely random at a noise level of 100%.

6.3.2 Real data

In the real data, as described in section 6.2.1, 225 unique landmarks are observed in the first two laps (the training data). In the operation phase, 107 of these landmarks are re-observed and used as input to the dynamic network. In the operation phase 114 new landmarks are detected, which are not processed by the current version of the model. Of the landmarks in the knowledge network 24% is ambiguous, that is, these landmarks are observed in more than one region in the training phase. The real data are quite challenging, because they contain noisy and erroneous observations, hold many ambiguous landmarks, and landmarks that are unequally distributed in the environment.

6.4 RESULTS

The results of the model on the simulated data are shown in Figure 6.3. Since the model keeps track of all hypotheses, there is a list of hypotheses with a decreasing activation value, not only a single winner. Hence, it is possible that the true place is not the best hypothesis, but the second best. Therefore, the performance of the model can be evaluated not only by comparing the true place to the best place hypothesis, but also to the top two or top three. Figure 6.3 depicts the best result (top one) for the context-based model and the signal-driven model, and the top two and three of the context-based model. The results of the signal-driven model are iden-
Figure 6.3: Results of the model tested on data with a varying amount of noise. The single best result of the context-based model and the signal-driven model are shown, and the top two and top three of the context-based model.

As expected, the signal-driven model performs at chance level. When an incorrect observation is made, the place prediction is also false. The context-based model performs better than the signal-driven model, especially for low amounts of noise (< 50%). In the experiment on the real data, of which the results are depicted in Figure 6.4, the context-based model also out-performs the signal-driven model. The difference between the score of the best hypothesis of both models is not very large, but consistent in multiple tests. However, the high scores on the

1 High levels of noise are not included in the figure, because the results are less meaningful if the noise is more prominent than the observations.
Figure 6.4: Results of the signal-driven and the context-based model on data collected by a moving robot.

top two and three are promising for future improvement.

It should be noted that the predictions of both models are based solely on visual observations, and odometric information is ignored. Therefore, the results of the two models can be compared by their performance on visual information, and we can show the advantage of the context-based model. If one would aim at a best possible robot localization, odometric information should be included.

6.5 Conclusion

We presented a model that dynamically manages a spreading activation network. This network represents the environment of an agent based on sensory information and knowledge of the environment. To test the applicability of the model in a real-world environment, we tested it on visual observations gathered by a mobile robot, with the goal to improve its localization. Learned knowledge about the environment of the robot is used to compute expectancies of its location. These expectancies are combined with instant observations to form a prediction of its location. Including expectancy in the prediction enhances the stability of the model, since it prevents unexpected landmarks from disrupting the place prediction.
The information about the environment is learned in a supervised training phase, and stored in a knowledge network, the long-term memory of the model. The short-term memory is represented by a dynamic network. Hypotheses in the dynamic network are more transient, because they represent the current state of the robot. The network and the deduced location prediction are updated when the robot gathers new evidence about its environment. The results of the experiments confirm that context evaluation improves the performance compared to signal-driven evaluation on both the simulated and the real data.

Although the context-based model outperforms the signal-driven model, the difference on the top one in the experiment on the real data is not very large. This can be explained by the fact that more than half of the landmarks that the robot encounters during the operation phase are new. Hence, the information on which the model can base its prediction is limited. Therefore, it will be useful to integrate an algorithm in the model that includes new landmarks in the knowledge network during the operation phase. For example, the growing-when-required (GWR) network of Marsland et al. (2002) adds new nodes to a network based on the (mis-)match between the data and the network. Such an algorithm would make it possible to learn new information during the operation phase. Furthermore, incremental learning can be used to update existing connections based on new observations.

Another possible improvement can be made in the determination of expectancies. In the current version of the model we only update the network when observations are made. This can pose problems to the model, especially when the data is not equally distributed over the environment, causing some places to be poorly represented by landmarks. Based on temporal and odometric information, expectancies of the path of the robot can be made even without observations.

In conclusion, the presented model can improve robot localization through context evaluation. It is computationally efficient and needs little memory storage. Therefore, it can be easily scaled to larger environments. Moreover, the model is general, because the sensory information in the model is not limited to visual observations. Hence, it can be used for state estimation in other domains (see chapter 5), or even combine information from different modalities to make predictions.