Towards Localization and Mapping of Autonomous Underwater Vehicles: A Survey

Technical Report: CES-515

Ling Chen and Huosheng Hu

School of Computer Science and Electronic Engineering
University of Essex, United Kingdom

Email: lcheno@essex.ac.uk, hhu@essex.ac.uk

September 2, 2011

Abstract

Autonomous Underwater Vehicles (AUVs) have been used for a huge number of tasks ranging from commercial, military and research areas etc, while the fundamental function of a successful AUV is its localization and mapping ability. This report aims to review the relevant elements of localization and mapping for AUVs. First, a brief introduction of the concept and the historical development of AUVs is given; then a relatively detailed description of the sensor system used for AUV navigation is provided. As the main part of the report, a comprehensive investigation of the simultaneous localization and mapping (SLAM) for AUVs are conducted, including its application examples. Finally a brief conclusion is summarized.

Contents

Towards Localisation and Mapping of Autonomous Underwater Vehicles: A survey

1 Introduction

Autonomous Underwater Vehicles (AUVs) are able to travel in underwater environments without inputs from a human operator[1], i.e. to control and manage themselves to accomplish missions given by human beings. As the ocean is closely related to environmental issues, resources, scientific and military tasks, the need for underwater robotic systems has become more apparent. Great efforts have been made in developing AUVs to operate in the unstructured and hazardous ocean environments [2]. Their applications can be categorized as follows.

- Commercial Application: Oil and gas industry uses AUVs to build detailed maps of the seafloor before they begin to deploy the infrastructures; AUVs with chemical sensors are used to monitor and detect the contamination state of specific sea areas.
- Military: A typical military mission for an AUV is to map an area to determine if there are any mines, or to monitor a protected area (such as a harbour) for new unidentified objects. AUVs are also employed in anti-submarine warfare, to aid in the detection of manned submarines.
- Research: Scientists in oceanographic and marine biology employ AUVs to dive into deep underwater to study marine life, to measure the concentration of various elements or compounds.

Without accurate localization, the AUV will get lost, not mention to realize other complex tasks. However, localization is never an independent issue because another issue called mapping always accompany with it, as only by referencing the environment (map) around the AUV, an AUV can localize itself. The map plays a key role in localisation unless a Global Position System (GPS) is deployed.

In other words, localization is a problem to localize the underwater vehicle by referencing the existing map around it, while mapping is a problem to build a map describing the environment through the knowledge of its own position. Therefore, localization and mapping is actually considered as “Chicken and Egg” problem. For decades, this problem has been attracting many researchers working on it, and at present it becomes a solved problem by the technique of Simultaneous Localization and Mapping (SLAM).

To the best knowledge of authors, SLAM has been successfully applied to localization and navigation of indoor, outdoor and undersea robots. Nevertheless, due to the highly unstructured and hazardous underwater and undersea environment, it is quite challenging to realize fully SLAM with high accuracy for AUVs. Consequently, the purpose of this survey is to investigate the relevant elements involved in localization and mapping of AUVs and previous work related to it, and to make suggestions for future research in this area.

The structure of this paper is organized as follows. Section 2 provides a brief introduction about the concept of AUVs and their historical development. Section 3 summarizes various sensors that are used for the localization and navigation of AUVs, in which the advantage and disadvantages of each type sensor is analyzed respectively. As the main part of the report, Section 4 presents comprehensive issues regarding the localization and mapping of AUVs, specifically on the definition of the localization and mapping problem, a comprehensive investigation of the solutions to the SLAM problem and the SLAM application examples for AUVs. Finally, a brief conclusion is summarized in Section 5.

2 Autonomous Underwater Vehicles

2.1 Development History

Research on AUVs can be dated back to 1960s, the successful invention of Rebikoff's SEA SPOOK and the University of Washington's SPURV(Self-propelled Underwater Research Vehicle) signified the beginning of the AUV's era. Soon after that, a plenty of AUVs showed up, for example, former SKAT (Soviet union), OSR-V (Japan), EAVE West?RUMIC and UFSS (U.S. Navy); EAVE EAST in University of New Hampshire, USA, as well as EPAULARD in France [3].

Unfortunately, AUVs developed relatively slowly in 1980s due to their big size, low efficiency and high cost, while the Remotely Operated Vehicles (ROVs) were beginning to gain in maturity. With the rapid advances and development of computer technology, energy storage and other relevant technology, the drawbacks stated above were conquered in 1990s. The AUV technology began mature. A brief chronological history of the AUV development is shown as follows [4].

- Prior to 1970s - Special application of AUVs

  Initial investigations were conducted into the utility of AUV systems, a few AUVs were built mostly to focus on very particular applications. There were not a huge number of papers published related to these efforts in literature.

- 1970 - 1980 - Explorer the potential of AUVs

  During these years, AUV technology was developing, a number of testbeds were built.

- 1980 - 1990 - Experiment with prototypes

  There were advances in technology reinforce development efforts, the Proof of Concept (POC) prototypes were developed/tested/used.

- 1990 - 2000 - Goal Driven Tech. Development

  Broader based funding of technology development was poured into the field, many AUVs were developed internationally, users were attracted by the technology.

- 2000 - 2010 - Commercial markets grow; Small bio-inspired AUV developed.

  First truly commercial products became available(See Figure 1); Several bio-inspired AUVs such as robotic fish in University of Essex[5](See Figure 2) and biomimetic robotic fish in Chinese Academy of Science[6].

2.2 Localisation and Navigation of AUV

The purpose of localisation is for the navigation, because only by knowing the current position can the AUV decide where to go and which action to take. Therefore, by saying navigation, most of its content includes localisation. The different methods which are currently used for AUV navigation can be grouped into three categories[7] in terms of the sensors that AUV use for navigation:

Figure 1: Commercial AUV

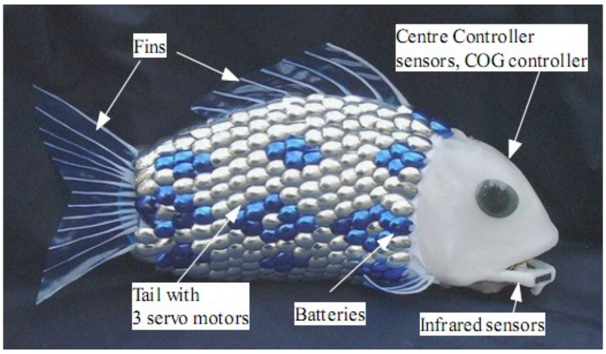


Figure 2: Bio-inspired robotic fish in University of Essex

1. Inertial navigation

   Inertial navigation uses gyroscopic sensors to detect the acceleration of the AUV. This is a significant improvement over dead reckoning and is often combined with a Doppler velocity log (DVL) which can measure the vehicle's relative velocity.

2. Acoustic navigation

   Acoustic navigation uses acoustic transponder beacons to allow the AUV to determine its position. The most common methods for AUV navigation are long baseline (LBL) which uses at least two, widely separated transponders and ultra-short baseline (USBL) which generally uses GPS-calibrated transponders on a single surface vessel.

3. Geophysical navigation

   Geophysical navigation uses physical features of the AUV environment to produce an estimate of the location of the AUV. These can be pre-existing or purposefully deployed features.

The detailed description of the sensors or sensor system narrated here will be provided in the following Section.

3 Sensors Used for Localization and Mapping

With eyes and ears, humans are able to localize and navigate anywhere around the world. Similarly, UAVs reuire various sensors for localization and navigation. These sensors include but not limited to GPS(Global Positioning System), INS(Inertial Navigation System), visual-based sensor, acoustic-based sensor and Underwater Acoustic Positioning System, etc.

3.1 GPS/INS

GPS, the acronym for Global Positioning System, is a space-based global navigation satellite system (GNSS) that provides location and time information in all weather, anywhere on or near the Earth, where there is an unobstructed line of sight to four or more GPS satellites. It is maintained by the United States government and is freely accessible by anyone with a GPS receiver[8]. An inertial navigation system (INS) is a navigation aid that uses accelerometers and gyroscopes to continuously calculate the position, orientation, and velocity (direction and speed of movement) of a moving object without the need for external references[9].

The GPS is capable of providing this accuracy if integrated with an INS to compensate for intermittent reception caused by either wave action or deliberate submergence. Therefore, GPS are usually integrated with INS to be applied in many areas, for example, integrated GPS/INS systems have been applied to aircraft and space shuttle guidance and navigation [10, 11, 12], balloon navigation[13], missile systems[14, 15], land vehicles[16], and mobile robots[17, 18]. In these applications, GPS data are continuously available in short intervals, and INS data are used to navigate between GPS fixes. Similar to these applications, integrated GPS/INS system can also be applied to AUVs working in shallow sea without long period of submergence.

When AUVs are surfaced, they take advantage of GPS to localize themselves accurately, while they are in underwater, INS replace GPS to localize though with relatively low accuracy compared to the circumstance on the surface. Yun *et al.* [19]developed a SANS(A Small Autonomous Underwater Vehicle Navigation System) system which employed GPS/INS as the localization and navigation sensor, exhibiting an encouraging degree of accuracy. Loebis *et al.*[20]described the implementation of an intelligent navigation system, based on the integrated use of the GPS and several INS sensors, for autonomous underwater vehicle (AUV) applications, differently from [19], they applied Kalman Filter to integrated the data from both GPS and INS. Zhao *et al.* [21] conducted the similar research to [20], integrating GPS/INS/DVL (Doppler Velocity Log) by applying Extended Kalman Filter (EKF).

Although GPS/INS integrated systems can achieve relatively high accuracy of localization, it is limited to shallow underwater environments with short period of working time, since INS has an accumulated error as the AUVs are moving underwater, the localization error will become quite large if it is not corrected by GPS for a long time.

3.2 Side-scan Sonar

Side-scan sonar (also sometimes called side scan sonar, sidescan sonar, side looking sonar, side-looking sonar, side imaging sonar, side-imaging sonar and bottom classification sonar) is a category of sonar systems that are used to efficiently create an image of large areas of the sea floor[22]. Figure 3 shows a working side-scan sonar.

Side-scan sonars can be used in marine or underwater fields for various purposes, such as pipeline detection and tracking[23], mine feature extraction[24] and geological interpretation[25] etc. One of the most significant applications of side-scan sonar is to help AUVs localize themselves. A variety of successful applications of side-scan sonar for localization of AUVs have been implemented all over the world. The general principle of how they apply it as the localization and mapping sensor is: firstly, collect image data from side-scan sonar; secondly, extract feature from the collected image data by some special algorithms; then finally, match the feature with *a priori* map to locate where the AUV is.

Langer *et al.* [26] used side-scan sonar for mapping. Rui *et al.* [27] create a system that builds a map of the environment using observations of landmarks extracted from side-scan sonar and uses that map and the dead-reckoning to estimate the AUV location. After the work done in [27], Rui *et al.* again uses a sidescan sonar to sense the environment to realize CML (Concurrent Localization and Mapping) in [28], in which the returns from the sonar are used to detect landmarks in the vehicles vicinity. These landmarks are used, in conjunction with a vehicle model, by the CML algorithm to concurrently build an absolute map of the environment and to localize the vehicle in absolute coordinates.

From what has been discussed above, we can make a conclusion about the merits and defects of the side-scan sensor used for localization and mapping. The merits include: the information provided by the side-scan sensor is rich, which may produce highly accuracy of localization. However, the defects exist in the following aspects: the volume of side-scan sonar is so big and the energy for powering it is so much that it is not appropriate to be installed on small AUV such as robotic fish stated above; the large volume of data needed to be processed adds a huge computational burden to the processors of side-scan sonar.

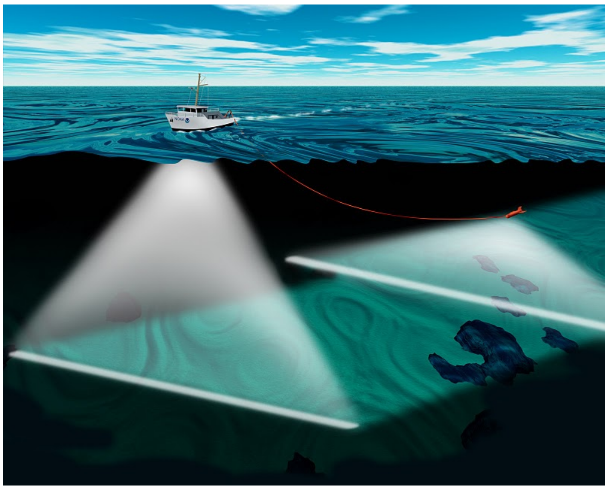


Figure 3: A side-scan sonar is working

3.3 Visual-based sensor

Camera and laser-based systems are the two main visual based sensors used for localization and mapping in underwater environments because of its low cost and rich information. Although video camera is limited to short range due to low visibility and lighting factors in underwater circumstances, it is widely applied by researchers conducting underwater localization and mapping experiment and practise. Salvi *et al.* [29] equipped calibrated stereo cameras on an AUV to implement Simultaneous Localization and Mapping (SLAM) by extended Kalman filters and unscented Kalman filters. Zhang *et al.* [30] built a AUV carrying the camera to localism its position by calculating the camera's viewpoint through looking at the artificial visual landmarks.

The position of landmarks is known in a world-centered coordinate system (WCCS). Two steps are then taken to realize the localization of the AUV: the first step is to calculate the 3D coordinates of feature points in a camera centred coordinate system (CCCS); the second step is to obtain a closed-form solution through the geometric transformations to map the 3D points from CCCS to WCCS. Carrera *et al.* [31] presented a vision-based localization system to estimate the position, orientation and velocity of an underwater robot in a structured environment. A down-looking camera is working with a coded pattern placed on the bottom of the water tank to provide map sensing information for localization, landmark detection and tracking.

As discussed above, the advantages of applying cameras to localize AUVs are: low cost and informative character; whereas the drawback is the fact that it is limited to a short range due to low visibility in underwater environments. Laser-based vision system is usually composed of laser projectors and a camera, which cooperate with each other to recognize the feature of specific objects. Compared to a single camera, the system is not subject to the low visibility and bad lighting condition of underwater environments, as the laser projectors can emit very powerful laser beam which can hardly be weakened by water.

Therefore, laser-based vision system can realize more accurate localization than a single camera. Unsurprisingly, there are application of this system for AUV localization. Karras *et al.* [32] proposed a ROV mounted with the sensor package consisting of two underwater laser pointers and a single CCD camera.The proposed system fuses data deriving from the projection of the laser pointers on the image plane and computer vision object tracking algorithms. The results deriving from the data fusion compose the position vector of the vehicle, with respect to the centre of the tracked object. The integration of the system was proved successful through the experimental procedure.

3.4 Underwater Acoustic Positioning System

Underwater Acoustic Positioning Systems are commonly used in a wide variety of underwater work, including oil and gas exploration, ocean sciences, salvage operations, marine archaeology, law enforcement and military activities[33]. Acoustic positioning systems measure positions relative to a framework of baseline stations, which must be deployed prior to operations.

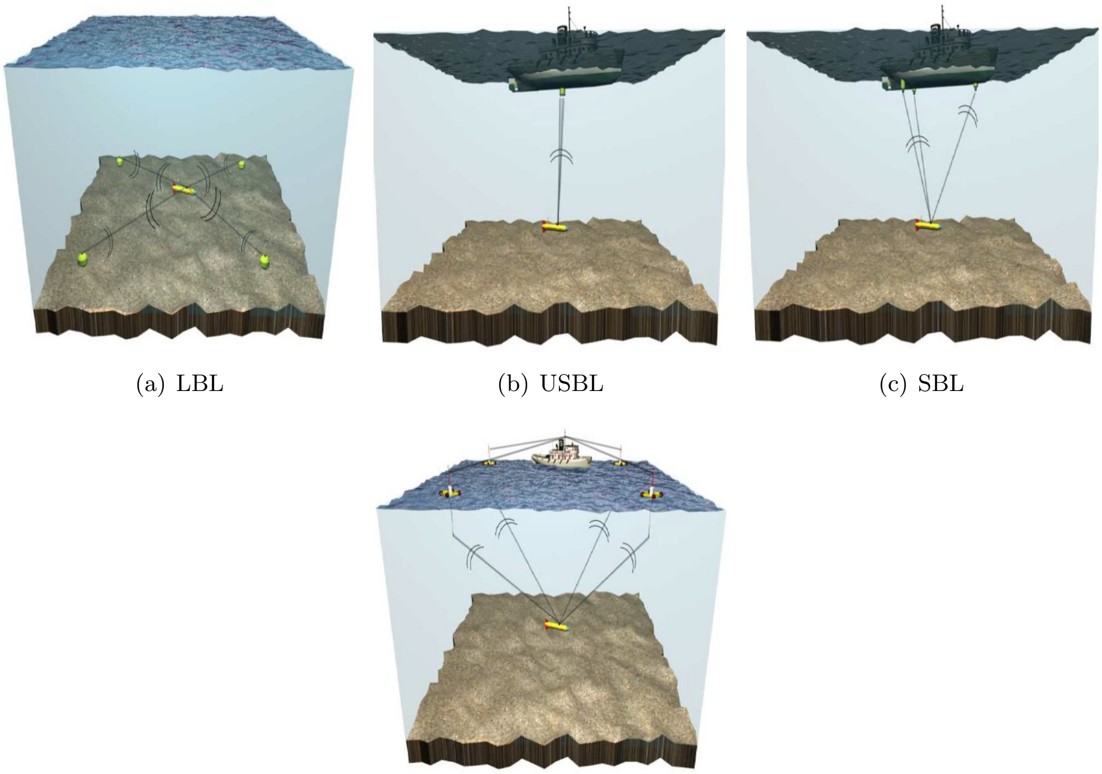
In the case of a long baseline (LBL) system(See Figure 4(a)), a set of three or more baseline transponders are deployed on the sea floor. The location of the baseline transponders either relative to each other or in global coordinates must then be measured precisely. Underwater acoustic positioning systems are generally categorised into four broad types or classes:

- Long Baseline (LBL) Systems

  Long baseline systems, as shown in Figure 4(a) above, use a sea-floor baseline transponder network. The transponders are typically mounted in the corners of the operations site. LBL systems yield very high accuracy of generally better than 1 m and sometimes as good as 0.01m along with very robust positions[34]. One of the typical applications of LBL for localization and navigation of AUV can be seen in[35], where Anibal Matos *et al.* successfully developed a LBL based navigation system for an AUV.

- Ultra Short Baseline (USBL) Systems

  USBL systems (see Figure 4(b)) and the related super short baseline (SSBL) systems rely on a small (ex. 230 mm across), tightly integrated transducer array that is typically mounted on the bottom end of a strong, rigid transducer pole which is installed either on the side or in some cases on the bottom of a surface vessel. Unlike LBL and SBL systems, which determine position by measuring multiple distances, the USBL transducer array is used to measure the target distance from the transducer pole by using signal run time, and the target direction by measuring the phase shift of the reply signal as seen by the individual elements of the transducer array. The combination of distance and direction fixes the position of the tracked target relative to the surface vessel. The disadvantage is that positioning accuracy and robustness is not as good as for LBL systems. The reason is that the fixed angle resolved by a USBL system translates to a larger position error at greater distance.



(a) LBL (b) USBL (c) SBL

(d) GIB

Figure 4: Classic underwater acoustic positioning systems. (a) Long Baseline LBL system. (b) Short Baseline SBL system. (c) Ultra-Short Baseline (USBL) system.(D) GPS Intelligent Buoys.

Therefore, USBL is generally integrated with other dead reckoning sensors such as DVL and INS for the accurate localization and navigation of AUV, by adopting filtering algorithms such as Kalman filter, Extended Kalman filter and Particle Filter etc. For example, Rigby *et al*. [36] improved navigation system which combines navigation data from a Doppler velocity Log (DVL) and an ultra-short baseline (USBL) acoustic tracking system to provide superior three-dimensional position estimates to the AUV; LI Shoujun *et al*. [37] integrated INS and USBL based on Kalman Filter to fulfil the deep-sea accurate navigation. Morgado *et al*. [38] presented a new Ultra-Short Baseline (USBL) tightly-coupled integration technique to enhance error estimation in low-cost strap down Inertial Navigation Systems (INSs) with application to underwater vehicles. Compared with a conventional loosely-coupled solution that consists of solving separately the triangulation and sensor fusion problems, the proposed technique enhances the position, orientation, and sensors biases estimates accuracy. The accuracy of USBL is much higher than that of LBL and SBL.

- Short Baseline (SBL) Systems

  Short baseline systems(see Figure 4(c)) use a baseline consisting of three or more individual sonar transducers that are connected by wire to a central control box. Accuracy depends on transducer spacing and mounting method. When a wider spacing is employed as when working from a large working barge or when operating from a dock or other fixed platform, the performance can be similar to LBL systems. In a contrast to the widely used application of USBL in underwater navigation, quite few SBL system was applied to this field.

- GPS Intelligent Buoys (GIB)

  GIB systems(see Figure 4(d)) are inverted LBL devices where the transducers are replaced by floating buoys, self-positioned by GPS. The tracked position is calculated in real-time at the surface from the Time-Of-Arrival (TOAs) of the acoustic signals sent by the underwater device, and acquired by the buoys. Such configuration allow fast, calibration-free deployment with an accuracy similar to LBL systems. At the opposite of LBL, SBL ou USBL systems, GIB systems use one-way acoustic signals from the emitter to the buoys, making it less sensible to surface or wall reflections. Alcocer *et al.* [39]selected GIB as the representative of a class of promising acoustic positioning systems and discussed theoretical and practical issues that must necessarily be taken into account during its operations at sea. Following this work, they moved on to a deeper research which is presented in [40], where they established an EKF GIB-based underwater positioning system, and tackled the problem of underwater target tracking in the framework of extended Kalman filtering by relying on a purely kinematic model of the target.

From what has been discussed above about the underwater acoustic positioning system, it is clear that the merits of the sensor system rest with the high accuracy of localization, and the drawbacks lie on a limited range, high complexity, expensive, and inflexibility (because of the deployment of the transponders).

In this section, we give a brief overview of various sensors used for localization and mapping of AUVs. Each type of sensors or sensory systems has its own merits and drawbacks. Therefore, the principle of choosing the most appropriate type of sensor should take into account the following factors: the working depth of AUV, the size of the AUV, the demanding accuracy of localization and the project budget etc.

4 Localisation and Mapping of AUV

Before we go deeper into the problem of localisation and mapping of AUVs, some general definitions should be given. As can be seen in Figure 5, there are 6 significant elements involved in this problem, whose specification is provided as follows:

- A discrete time index $k = 1, 2, \cdots$.
- x_k: The true location of the vehicle at a discrete time k.
- u_k: A control vector, assumed known, and applied at time $k-1$ to drive the vehicle fromx_{k-1} to x_k at time k.
- m_i: The true location or parametrisation of the i^{th} landmark.
- $z_{k,i}$: An observation (measurement) of the i^{th} landmark taken from a location x_k at time k.
- z_k: The (generic) observation (of one or more landmarks) taken at time k.

In addition to these elements, the following sets are also defined:

- The history of states: $X^k = \{x_0, x_1, \cdots, x_k\} = \{X^{k-1}, x_k\}$.
- The history of control inputs: $U^k = \{u_1, u_2, \cdots, u_k\} = \{U^{k-1}, u_k\}$.
- The set of all landmarks: $m = \{m_1, m_2, \cdots, m_M\}$.
- The history of observations: $Z^k = \{z_1, z_2, \cdots, z_k\} = \{Z^{k-1}, z_k\}$.

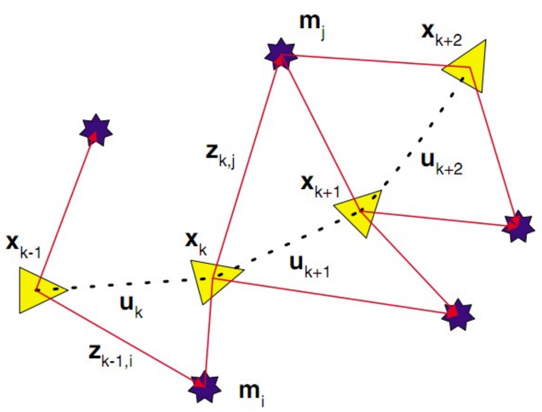


Figure 5: Elements involved in localisation and mapping problem. Stars represent the landmarks, triangles represent the vehicle and the dashed line represents the trajectory of the vehicle.

4.1 The Localisation Problem

As shown in Figure 6, the localisation problem can be described as to make inferences about the unknown vehicle locations X^k under the control of a sequence of control actions U^k by taking the observation z_k, given a known *a priori* map which may be a geometric map, a map of landmarks and a map of occupancy. Due to the inevitable uncertainty or measurement error, the estimation of the vehicle location diverges from the true position of the vehicle.

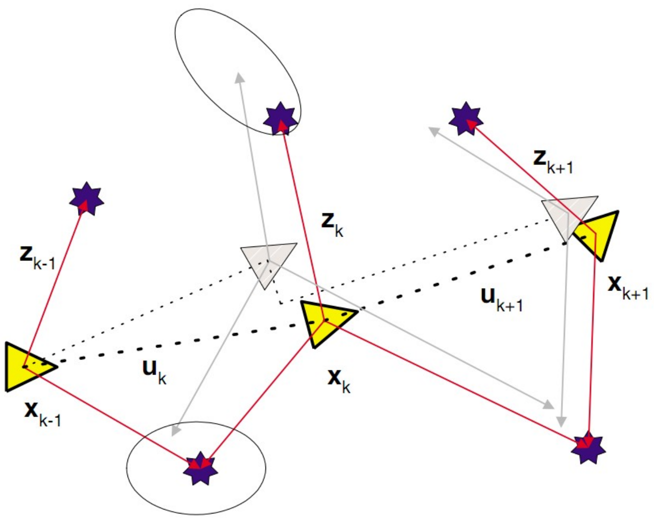


Figure 6: The localisation problem. Stars represent the landmarks, the yellow triangles represent the true position of the vehicle, the grey triangles represent the estimated of inferred position, and the dashed line represents the trajectory of the vehicle.

4.2 The Mapping Problem

A major step towards achieving the true autonomy of mobile robots is the ability to independently extract and recognize useful knowledge from the environment. For the navigation purpose, this knowledge is commonly known as a map. As can be seen from Figure 7, the mapping problem can be described as to make inferences about the map m, which may be a geometric map, a map of landmarks or a map of occupancy, given the vehicle locations X^k by some independent means such as GPS. Similar to the localisation problem, due to the unavoidable uncertainty of the observation, the estimation of the map slightly diverges from the true position of the map.

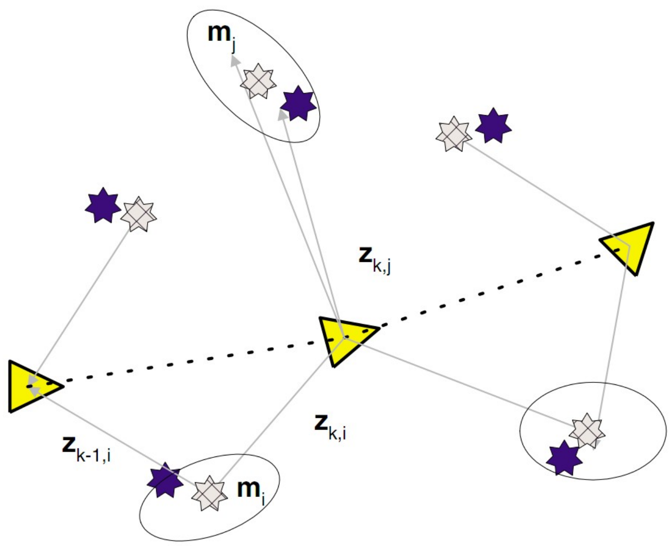


Figure 7: The mapping problem. The purple stars represent the true position of landmarks or map, the grey stars represent the estimated position of landmarks, triangles are vehicles, and the dashed line represents the trajectory of the vehicle.

Basically, the method of map representation can be categorized into three classes: the $grid-based$ map, $feature-based$ map and the *topological* map.

4.2.1 Grid-based Map

Grid based approaches originally proposed by Moravec and Elfes [41, 42, 43, 44] represent the robot's environment by dividing the workspace into evenly spaced grids. Each grid cell represents a part of the world and has a probability attached to it which indicates whether it is occupied or not. This probability indicates whether the centre of the robot can move into the centre of that particular grid cell or not.

To build grid maps, sensor readings should be translated into occupancy values for each grid cell. Thrun *et al.* in [45] and Van Dam *et al.* in [46] proposed the back propagation neural network (see Figure 8) to map sensor data to occupancy values for each grid cell. As can be seen from Figure 8, the input to the network consists of the four sensor readings closest to the cell$< x, y >$, together with two values that encode the cell location in polar coordinates (see Figure 8 and Figure 9). The output is a binary value 0 indicating an unoccupied cell (i.e. no obstacle) and 1 indicating an occupied cell (presence of obstacles). Figure 10 shows an example of a grid based map.

In order to obtain a consistent map, sensor interpretations are integrated over time in updating the grid cells. To do this, Moravec and Elfes [47] and Cho [48] used Bayesian rule, while Oriolo *et al.* [49], and Fabrizi & Saffioti [50] used fuzzy logic.

The advantages of using grid-based maps are:

- The maps are easy to build, represent and maintain.
- Recognition of places is non-ambiguous and independent of view point.
- The map allows for easy computation of shortest paths.

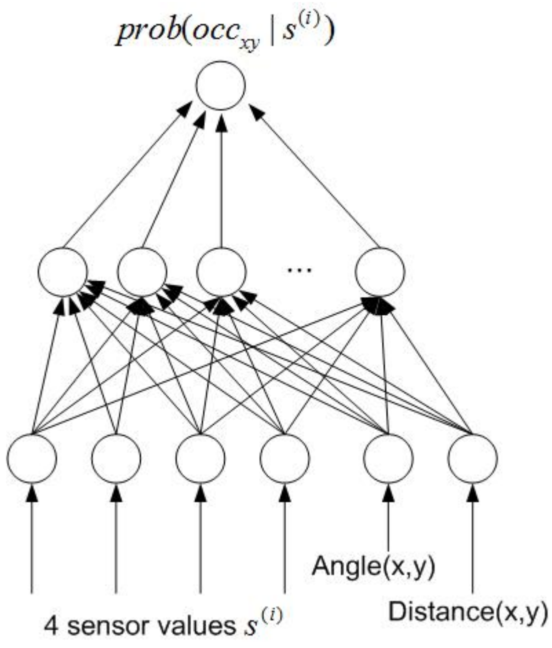


Figure 8: Neural Network maps sensor measurement to occupancy probability[45]

While the disadvantages rest with:

- Computationally heavy for grid updating routine.
- The maps consume a lot of memory space as the environment size increases.
- The maps are inefficient for planning.
- Require accurate determination of the robots position.
- The map is a poor interface for most symbolic problem solvers.

4.2.2 Feature-based Map

Most environments usually contain one or more distinctive features which can be parametrised by metrics such as width, length, position, shapes or by colour, etc. Feature based maps are built by descriptions of these features of the environment. In [52], Samuel *et al.* proposed an line fitting algorithm to extract line feature from raw range data (see Figure 11). The simplest feature based map m typically contains a list of N features, probably of the same type or structure, which describes their position within the within the environment, and can be described as $m = \{m_1, \cdots, m_N\}$.

In [53], Cowley used line segments to represent the environment, where each segment has an attribute representing its degree of confidence. Chatila *et al.* [54] used a polygonal map to predict readings from a laser sensor. Kuc & Siegel [55] implemented map building by considering lines, corners and edges as suitable candidates for geometric beacons. Johan *et al.* [56] developed an extension to the Hough transform by range weighting. The *range weighted Hough transform* is used for extracting line segments from the environment for map making and for implementing Kalman filter based navigation. Lamon *et al.* [57] proposed a method for creating unique identifiers by combining vertical lines and colour patches extracted from vision, to encode *fingerprint* sequences as environmental features in the map. The advantage of describing sensor information in terms of an uncertain parametric function is that the geometric description itself can be transformed between different coordinate systems and different object

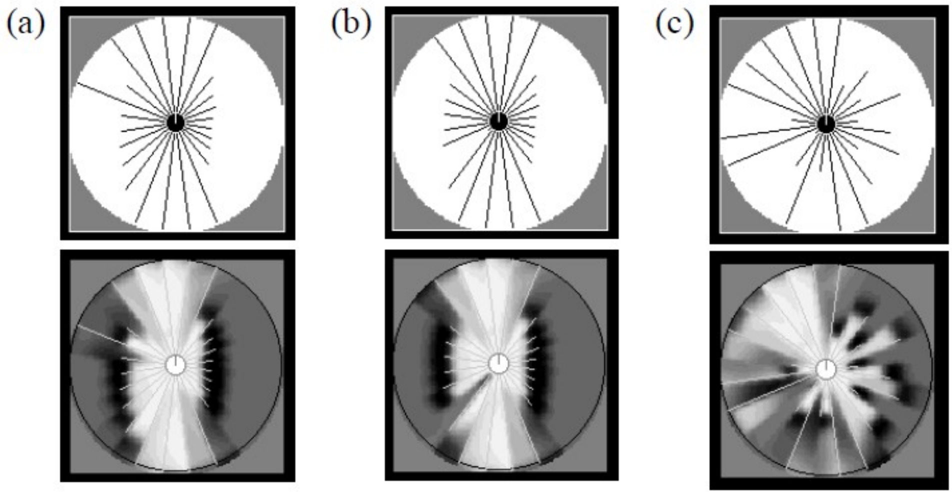


Figure 9: Sensor interpretation:Three sample sonar scans (top row) and local occupancy maps (bottom row), as generated y the neural network. Bright regions indicate free space, dark regions indicate walls and obstacles (enlarged by robot diameter)[45]

representations, providing a simple but effective means of communicating information between sensors[58].

4.2.3 Topological Maps

Topological maps are built on top of the grid-based maps. The key idea is simple but very effective. The free-space of a grid-based map is partitioned into a small number of regions, separated by critical lines that correspond to narrow passages such as doorways. The partitioned map is then mapped into a isomorphic graph[51]. This method applies a connected graph structure technique to represent the environment, graph nodes correspond to places and edges correspond to connections between places. The map is defined by the structure of the environment, and a place is an area that is a functional or a topological unit[59]. Figure 12 shows the process of transforming a grid-based map into a topological map.

Brooks [60] built a topological map using vision, where building blocks are called freeways (edges) and meadows (nodes). Kortenkamp, Kuipers. and others [61, 62, 63] also represent robot environments by graphs.

Advantages of the topological approach can be concluded as:

- The map can be constructed without consideration of the exact geographical relationship between nodes, and so does not require accurate determination of the robot's position. All that is required is a knowledge of which edge the robot is getting into when leaving a particular node. This means that, every node must be distinguishable from others in its immediate neighbourhood by some criterions as defined by sensory inputs.
- The map permits efficient planning, and its resolution depends on the complexity of the environment.
- It is a convenient representation for symbolic planning and problem solving.

whereas the disadvantages lie in:

- Difficulty in disambiguating between places where two places look alike. *"Since sensory input usually depends strongly on the viewpoint of the robot, topological approaches may fail to recognize geometrically nearby places even in static environments, making it difficult*

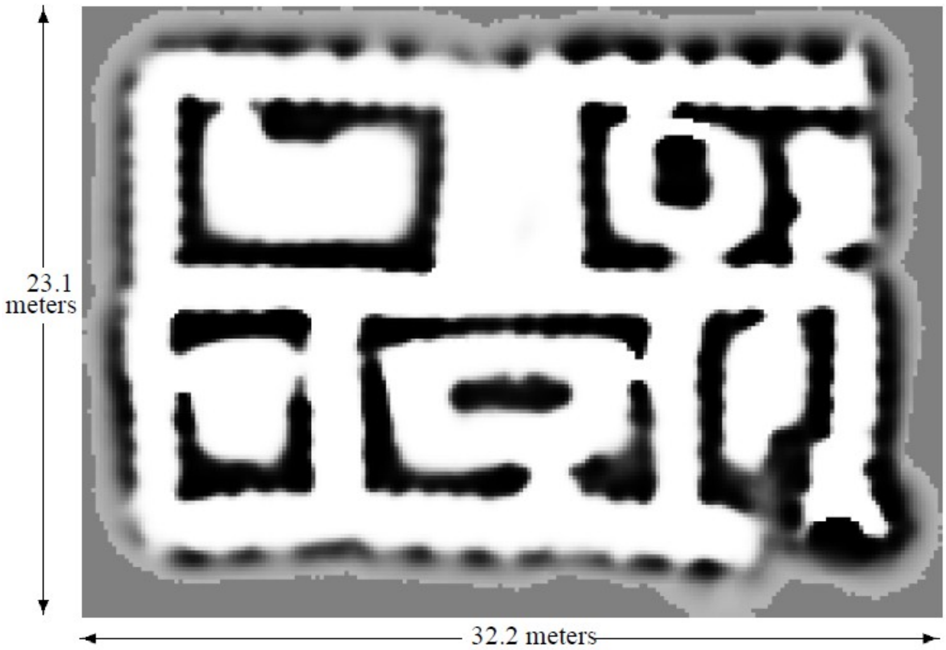


Figure 10: Grid-based map, constructed at the 1994 AAAI autonomous mobile robot competition[51].

to construct large scale maps, particularly if sensor information is highly ambiguous" by Sebastian Thrun [64].

- A map may yield sub optional paths

4.3 The Simultaneous Localisation and Mapping Problem

The simultaneous localisation and mapping (SLAM) problem is, for a vehicle to be placed at an unknown location in an unknown environment, to incrementally build a consistent map of this environment while simultaneously determining its location within this map[65]. As shown in Figure 13, using the definitions stated at the beginning of this section, the SLAM problem can also be described as to build the map m and at the same time make inferences about the location of the vehicle X^k under the conditions that no information about m is provided, and the initial location x_0 is assumed known (the origin) and that the sequence of control actions U_k is given.

Uncertainty arises if the robot lacks critical information for carrying out its task. It arises from five different factors: environment, sensors, robot it self, models and computation. All these factors will result in the errors between the true position and estimated position of both the map and the vehicle.

Smith *et al.* [66] published the first consistent solution to the SLAM problem where they employ estimation and the Kalman filter. Early experiments were later carried out by Moutarlier *et al.* [67] where the approach used by Smith *et al.* was modified; the problem of bias in the filter was also addressed. Experiments were also carried out by Leonard *et al.* [68]. The structure of the SLAM problem, the convergence result and the coining of the acronym SLAM was first presented in a mobile robotics survey paper presented at the 1995 International Symposium on Robotics Research[69]. Csorba in [70] developed the essential theory on convergence and many of the initial results. The complexity of the problem is as a result of the fact that both the position of a feature end the vehicle are estimated from one relative position, which can be likened to a single algebraic equation of two unknowns.

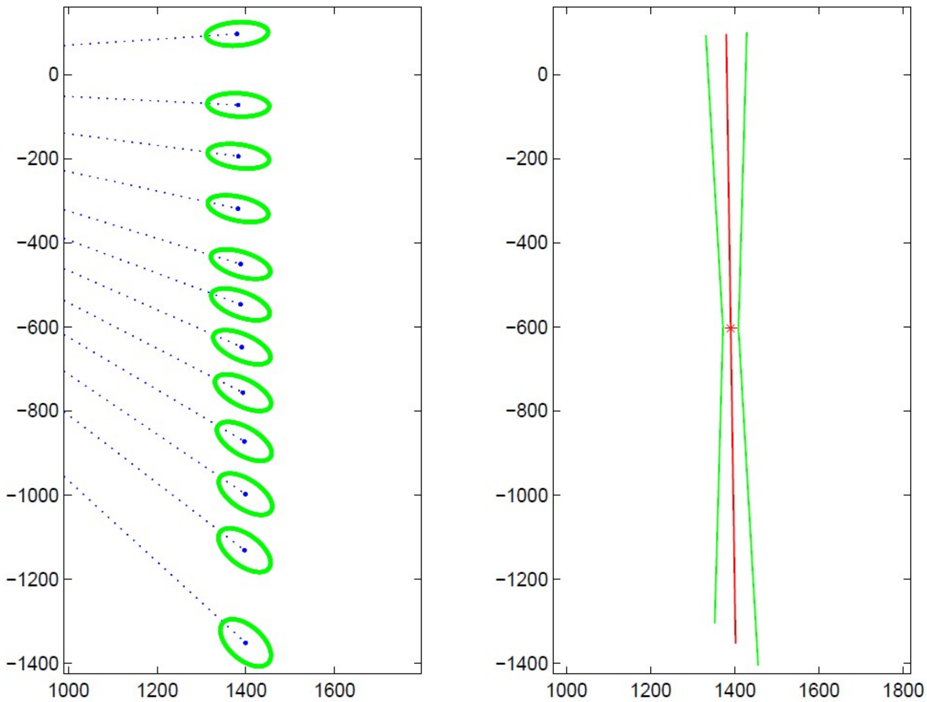


Figure 11: Example of line segment fit: data points (left) and fitted line with a representation of its uncertainty (right)[52]

4.3.1 Probabilistic Model for SLAM

As stated above, uncertainty exists almost every part of the robot control. We can naturally choose probabilistic model to describe and even give solution to SLAM problem. Actually probabilistic models of sensing and motion are the most widely used method of quantifying uncertainty. According to Durrant-Whyte *et al*. [65], for probabilistic form, the SLAM problem requires that the probability distribution

$$P(x_k, m|Z_{0:k}, U_{0:k}, x_0) \tag{1}$$

be computed for all times k. This probability distribution describes the joint posterior density of the landmark locations and vehicle state (at time k) given the recorded observations and control inputs up to and including time k together with the initial state of the vehicle.

In general, a recursive solution to the SLAM problem is desirable. Starting with an estimate for the distribution $P(x_{k-1}, m|Z_{0:k-1}, U_{0:k-1})$ at time $k-1$, the joint posterior, following a control u_k and observation z_k, is computed using Bayes theorem. This computation requires that a state transition model and an observation model are defined describing the effect of the control input and observation respectively. The *observation model* describes the probability of making an observation z_k when the vehicle location and landmark locations are known and is generally described in the form

$$P(z_k|x_k, m). \tag{2}$$

It is reasonable to assume that once the vehicle location and map are defined, observations are conditionally independent given the map and the current vehicle state. The *motion model* for the vehicle can be described in terms of a probability distribution on state transitions in the

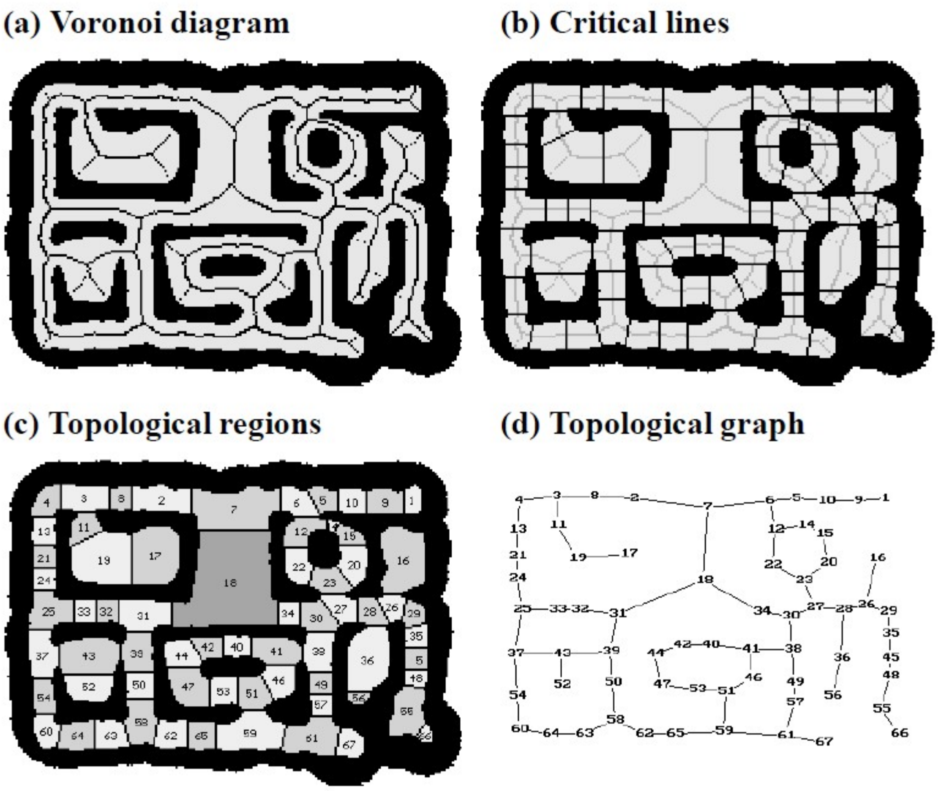


Figure 12: Extracting the topological graph from the map depicted in Figure 10: (a) Voronoi diagram, (b) Critical points and lines, (c) regions, and (d) the final graph.[51]

form

$$P(x_k|x_{k-1}, u_k). \tag{3}$$

That is, the state transition is assumed to be a Markov process in which the next state x_k depends only on the immediately preceding state x_{k-1} and the applied control u_k and is independent of both the observations and the map.

The SLAM algorithm is now implemented in a standard two-step recursive (sequential) prediction (time-update) correction (measurement-update) form:

- **Time-update**

$$P(x_k, m|Z_{0:k-1}, U_{0:k}, x_0) = \int P(x_k|x_{k-1}, u_k) \times P(x_{k-1}, m|Z_{0:k-1}, U_{0:k-1}, x_0)dx_{k-1} \tag{4}$$

- **Measurement-update**

$$P(x_k, m|Z_{0:k}, U_{0:k}, x_0) = \frac{P(z_k|x_k, m)P(x_k, m|Z_{0:k-1}, U_{0:k}, x_0)}{P(z_k|Z_{0:k-1}, U_{0:k})} \tag{5}$$

Both Equation 4 and Equation 5 provide a recursive procedure for calculating the joint posterior $P(x_k, m|Z_{0:k}, U_{0:k}, x_0)$ for the robot state x_k and map m at a time k based on all observations $Z_{0:k}$ and all control inputs $U_{0:k}$ up to and including time k. The recursion is a function of a vehicle model $P(x_k|x_{k-1}, u_k)$ and an observation model $P(z_k|x_k, m)$.

It is worth noting that the mapping problem can be formulated as computing the conditional

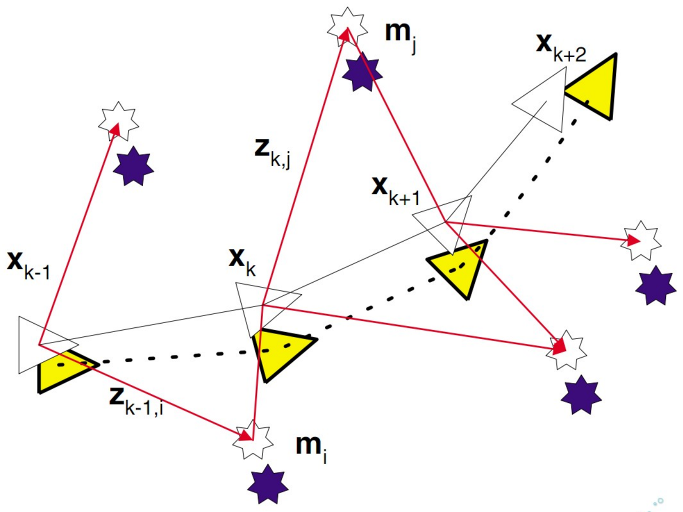


Figure 13: The SLAM problem. The purple stars and the yellow triangles represent the true position of the map and the vehicle respectively. The white stars and triangles represent the estimated position of both the map and the vehicle, and the dashed line represents the trajectory of the vehicle.

density $P(m|X_{0:k}, Z_{0:k}, U_{0:k})$, which assumes that the location of the vehicle x_k is known(or at least deterministic) at all times, subject to knowledge of initial location. Similarly, the localisation problem can be formulated as computing the probability distribution $P(x_k|Z_{0:k}, U_{0:k}, m)$, which assumes that the landmark locations are known with certainty, and the objective is to compute an estimate of vehicle location with respect to these landmarks.

4.4 Solution to SLAM

4.4.1 Extended Kalman Filter

The Extended Kalman filter (EKF) is a linear recursive estimator for systems described by non-linear process models and/or observation models. As its name shows, EKF is an improved filter on the basis of Kalman filter that addresses the general problem of estimating the state of a discrete-time controlled process governed by a linear stochastic difference equation. EKF aims to tackle the scenario where the process to be estimated and (or) the measurement relationship to the process is non-linear.

According to Durrant-Whyte *et al.* [65], the basis for the EKF-SLAM method is to describe the vehicle motion in the form

$$P(x_k|x_{k-1}, u_k) \Leftrightarrow x_k = f(x_{k-1}, u_k) + w_k, \tag{6}$$

where $f(\cdot)$ models vehicle kinematics and where w_k are additive, zero mean uncorrelated Gaussian motion disturbances with covariance Q_k. The observation model is described in the form

$$P(z_k|x_k, m) \Leftrightarrow z_k = h(x_k, m) + v_k, \tag{7}$$

where $h(\cdot)$ describes the geometry of the observation and where v_k are additive, zero mean uncorrelated Gaussian observation errors with covariance R_k.

Time-update

$$\hat{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_k) \tag{8}$$

$$P_{xx,k|k-1} = \nabla f P_{xx,k-1|k-1} \nabla f^T + Q_k \tag{9}$$

where ∇f is the Jacobian of f evaluated at the estimate $\hat{x}_{k-1|k-1}$. There is generally no need to perform a time-update for stationary landmarks.

Observation-update

$$\begin{bmatrix} \hat{x}_{k|k} \\ \hat{m}_k \end{bmatrix} = \begin{bmatrix} \hat{x}_{k|k-1} \, \hat{m}_{k-1} \end{bmatrix} + W_k(z_k - h(\hat{x}_{k|k-1}, \hat{m}_{k-1})) \tag{10}$$

$$P_{k|k} = P_{k|k-1} - W_k S_k W_k^T \tag{11}$$

where

$$S_k = \nabla h P_{k|k-1} \nabla h^T + R_k$$

$$W_K = P_{k|k-1} \nabla h^T S_k^{-1}$$

and where ∇h is the Jacobian of h evaluated at $\hat{x}_{k|k-1}$ and $\hat{m}_{k-1}$.

Equations (8), (9), (10) and (11) compose the essence of the EKF algorithm. EKF is by far the most widely used algorithm for problems in localisation, mapping, and navigation, and it is also considered as the basis for most current SLAM algorithms. The following two algorithms (UKF and PF) are exactly developed on the basis of EKF.

David Ribas *et al.*[71] chose an extended Kalman filter (EKF)Cbased implementation of the stochastic map to perform SLAM for an AUV. The algorithm relies on a mechanically scanned imaging sonar (MSIS) for feature extraction. A mechanically scanned imaging sonar is used to obtain information about the location of vertical planar structures present in such environments. The obtained information is incorporated into a feature-based simultaneous localisation and mapping (SLAM) algorithm running an extended Kalman filter. Mahon, I. *et al.*[72] also applied EKF-SLAM to select and track regions of the environment that may be used as features to estimate the vehicle's motion.

Although EKF is the most widely used algorithm in localisation and mapping field, it still has several disadvantages, which are:

- Unlike its linear counterpart, the extended Kalman filter in general is not an optimal estimator (of course it is optimal if the measurement and the state transition model are both linear, as in that case the extended Kalman filter is identical to the regular one).
- In addition, if the initial estimate of the state is wrong, or if the process is modelled incorrectly, the filter may quickly diverge, owing to its linearisation.
- Another problem with the extended Kalman filter is that the estimated covariance matrix tends to underestimate the true covariance matrix and therefore risks becoming inconsistent in the statistical sense without the addition of "stabilising noise"[73].

4.4.2 Unscented Kalman Filter

The extended Kalman filter is an minimum mean squared error (MMSE) estimator that approximates the probability distribution function $P(x_k|Z_k)$, based on Taylor series expansion of the process and measurement functions around the estimates $\hat{x}_{k|k}$ of the state $x(k)$. Because of the first order expansion used in obtaining the EKF, large errors may be introduced in the posterior estimates, especially when the model is highly nonlinear and the local linearity assumption breaks down, and also may lead to sub-optimal performance and sometimes divergence of the filter[74]. The $unscented\ Kalman\ filter$ (UKF), a recursive MMSE was first proposed by Julier & Uhlmann[75] in an attempt to overcome the adverse effects of the first order approximations in EKF, by utilising the $Unscented\ Transform$ (UT). UT is used to calculate the statistics of a random variable that undergoes a nonlinear transformation and builds on the principle that it is easier to approximate a probability distribution instead of approximating nonlinear functions.

The Unscented Transform Consider a n_x dimensional random variable x that passes through an arbitrary nonlinear function $g : \mathbb{R}^{n_x} \to \mathbb{R}^{n_y}$ to generate y,

$$y = g(x) \tag{12}$$

Given the known mean $\bar{x}$ and covariance P_x of x, we calculate the mean and covariance of y by using the following procedures.

- Step 1:

  A set of $2n_x + 1$ weighted samples or $sigma\,points\, S_i = \{W_i, \chi_i\}$ are chosen deterministically to capture the true mean and covariance of the prior x. Based on the following selection scheme[75]:

$$if\ i = 0 \qquad \chi_0 = \bar{x} \qquad W_0 = k/(n_x + k) \tag{13}$$

$$for\ i = 1, \cdots, n_x \qquad \chi_i = \bar{x} + \left(\sqrt{(n_x + k)P_x}\right)_i \qquad W_i = 1/2(n_x + k) \tag{14}$$

$$for\ i = n_x + 1, \cdots, 2n_x \qquad \chi_i = \bar{x} - \left(\sqrt{(n_x + k)P_x}\right)_i \qquad W_i = 1/2(n_x + k) \tag{15}$$

  where k is a scaling parameter and $\left(\sqrt{(n_x + k)P_x}\right)_i$ is ith row or column of the matrix square root of $(n_x + k)$, W_i is the weight associated with ith point $\epsilon\Sigma_{i=0}^{2n_x} W_i = 1$

- Step 2:

  Propagate each sigma point χ_i through the nonlinear function to obtain

$$y_i = g(\chi_i) \qquad i = 0, \cdots, 2n_x$$

- Step 3:

  Compute the estimated mean and covariance of y as follows

$$\bar{y} = \sum_{i=0}^{2n_x} W_i y_i \tag{16}$$

$$P_y = \sum_{i=0}^{2n} W_i (y_i - \bar{y})(y_i - \bar{y})^T \tag{17}$$

These estimates are accurate to the second order Taylor series expansion of $g(x)$ for any nonlinear function, and accurate to the third order for Gaussian priors. Errors that are introduced in the third and higher order terms are scaled by the choice of k.

A property of UT is that as the state space dimension increases, the spread of the sigma points increases. The first and second moments of the prior are still well captured but, if the nonlinearities are very serious then there could be significant difficulties in the estimation. The problem is addressed by the choice of the scaling factor k. By variating the value of k, sigma points can be scaled towards or away from $\hat{x}$. The $Scaled\ Unscented\ Transform$ (SUT) is used to replace the original set of sigma points with a transformed set.

The implementation of the UKF is formulated as follows:

1. Initialisation

$$\begin{aligned}
\hat{x}(0|0) &= E[x(0)] \\
P(0|0) &= E\left[x(0) - \hat{x}(0|0)(x(0) - \hat{x}(0|0))^T\right] \\
\hat{x}^a(0|0) &= E\left[x^a(0|0)\right] = \left[\hat{x}(0|0)^T \ 0 \ 0\right] \\
P^a(0|0) &= E\left[\left(x^a(0) - \hat{x}^a(0|0)\right)\left(x^a(0) - \hat{x}^a(0|0)\right)^T\right] \\
&= \begin{bmatrix} P(0|0) & 0 & 0 \\ 0 & Q & 0 \\ 0 & 0 & R \end{bmatrix}
\end{aligned}$$

2. For $i \in 1, \cdots, \infty$,

3. Calculate sigma points

$$\chi^a(k|k) = \left[\hat{x}^a(k|k) \ \hat{x}^a(k|k) \pm \sqrt{(n_a + \lambda)P_1^a(k|k)}\right] \tag{18}$$

4. Time Update (prediction)

$$\chi^x(k+1|k) = f\left(\chi^x(k), \chi^v(k)\right) \tag{19}$$

$$\hat{x}(k+1|k) = \sum_{i=0}^{2n_a} W_i^{(m)} \chi_i^x(k+1|k) \tag{20}$$

$$P(k+1|k) = \sum_{i=0}^{2n_a} W_i^{(c)} \left[\chi_i^x(k+1|k) - \hat{x}(k+1|k)\right]\left[\chi_i^x(k+1|k) - \hat{x}(k+1|k)\right]^T \tag{21}$$

$$y(k+1|k) = h(\chi^x(k+1|k), \chi^n(k)) \tag{22}$$

$$\hat{y}(k+1|k) = \sum_{i=0}^{2n_a} W_i^{(m)} y_i(k+1|k) \tag{23}$$

5. Measurement Update (correction)

$$P_{\tilde{y}\tilde{y}}(k+1) = \sum_{i=0}^{2n_a} W_i^(c) \left[y_i(k+1|k) - \hat{y}(k+1|k)\right] \left[\left[y_i(k+1|k) - \hat{y}(k+1|k)\right]\right] \tag{24}$$

$$P_{xy}(k+1) = \sum_{i=0}^{2n_a} W_i^{(c)} \left[\chi_i(k+1|k) - \hat{x}(k+1|k)\right] \left[y_i(k+1|k) - \hat{y}(k+1|k)\right]^T \tag{25}$$

Kalman gain

$$K(K+1) = P_{xy}(k+1)P_{\tilde{y}\tilde{y}}^{-1}(k+1) \tag{26}$$

Updated Estimate

$$\hat{x}(k+1|k+1) = \hat{x}(k+1|k) + K(k+1)(y(k+1) - \hat{y}(k+1)) \tag{27}$$

Updated Covariance

$$P(k+1|k+1) = P(k+1|k) - K(k+1)P_{\tilde{y}\tilde{y}}(k+1)K^T(k+1) \tag{28}$$

Lian *et al.*[76] applied UKF(unscented Kalman filter) to the AUV integrated navigation. As compared with the Dead Reckoning and EKF, the simulation result shows that the UKF is more accurate than the first two and with comparable compute cost. LIU *et al.*[77] presented an improved unscented Kalman filter(UKF) algorithm to meet the stability,precision and real-time performance of navigation system for AUV. A simplified UKF integrated navigation algorithm based on spherical simplex sampling transformation is designed for the additive and complex noise model of the system. Simulation results show that this algorithm can effectively reduce the computational complexity and improve the efficiency of the navigation system without loss of filtering accuracy compared with the traditional UKF algorithm using scaled symmetric sampling.

4.4.3 Particle Filter

The Kalman Filter and the EKF represent probability distributions using a parameterized model (a multivariate Gaussian). Particle filters, on the other hand, represent distributions using a finite set of sample states, or "particles." Regions of high probability contain a high density of particles, whereas regions of low probability contain few or no particles. Given enough samples, this non-parametric representation can approximate arbitrarily complex, multi-modal distributions[78]. As particle filters have high efficiency of computing when used for SLAM, they are also known as FastSLAM.

The joint SLAM state may be factored into a vehicle component and a conditional map component:

$$P(X_{0:k}, m|Z_{0:k}, U_{0:k}, x_0) = P(m|X_{0:k}, Z_{0:k})P(X_{0:k}|Z_{0:k}, U_{0:k}, x_0) \tag{29}$$

Here the probability distribution is on the trajectory $X_{0:k}$ rather than signle pose x_k because, when conditioned on the trajectory, the map landmarks become independent. This is a key property of Particle Filter (PF) and reason for its speed.

The essential structure of PF is a Rao-Blackwellized state, where the trajectory is represented by weighted samples and the map is computed analytically. Thus the joint distribution, at time

k, is represented by the set $\left\{w_k^{(i)}, X_{0:k}^{(i)}, P(m|X_{0:k}^{(i)}, Z_{0:k})\right\}_i^N$, where the map accompanying each particle is composed of independent Gaussian distributions

$$P(m|X_{0:k}^{(i)}, Z_{0:k}) = \prod_j^M P(m_j|X_{0:k}^{(i)}, Z_{0:k}).$$

The general form of a particle filter for SLAM is as follows. We assume that, at time $k-1$, the joint state is represented by $\left\{w_k^{(i)}, X_{0:k-1}^{(i)}, P(m|X_{0:k-1}^{(i)}, Z_{0:k-1})\right\}_i^N$

1. For each particle, compute a proposal distribution, conditioned on the specific particle history, and draw a sample from it

$$x_k^{(i)} \sim \pi(x_k|X_{0:k-1}^{(i)}, Z_{0:k}, u_k). \tag{30}$$

   This new sample is (implicitly) joined to the particle history $X_{0:k}^{(i)} \triangleq \{X_{0:k-1}^{(i)}, Z_{0:k}, u_k\}$.

2. Weight samples according to the importance function

$$w_k^{(i)} = w_{k-1}^{(i)} \frac{P\left(z_k|X_{0:k}^{(i)}, Z_{0:k-1}\right) P\left(x_k^{(i)}|x_{k-1}^{(i)}, u_k\right)}{\pi\left(x_k^{(i)}|X_{0:k-1}^{(i)}, Z_{0:k}, u_k\right)}. \tag{31}$$

   The numerator terms of this equation are the observation model and the motion model, respectively. The former differs from (2) because R-B requires dependency on the map be marginalised away.

$$P(z_k|X_{0:k}, Z_{0:k-1}) = \int P(z_k|x_k, m)P(m|X_{0:k-1}, Z_{0:k-1})dm. \tag{32}$$

3. if necessary, perform resampling. (When best to instigate resampling is an open problem. Some implementations resample every time-step, others after a fixed number of time-steps, and others once the weight variance exceeds a threshold.) Resampling is accomplished by selecting particles, with replacement, from the set $\left\{X_{0:k}^{(i)}\right\}_i^N$, including their associated maps, with probability of selection proportional to $w_k^{(i)}$. Selected particles are given uniform wight, $w_k^{(i)} = 1/N$.

4. For each particle, perform an EKF update on the observed landmarks as a sample mapping operation with known vehicle pose.

There are two versions of FastSLAM in the literature, FastSLAM 1.0 (First proposed in [79] by M. Montemerlo *et al.*) and FastSLAM 2.0 (First proposed in [80] by the same group of people with FastSLAM 1.0), differ only in terms of the form of their proposal distribution (Step 1) and, consequently, in their importance weight (Step 2). FastSLAM 2.0 is by far the more efficient solution. For FastSLAM 1.0, the proposal distribution is the motion model

$$x_k^{(i)} \sim P\left(x_k|x_{k-1}^{(i)}, u_k\right). \tag{33}$$

Therefore, from (31), the samples are weighted according to the marginalised observation model.

$$w_k^{(i)} = w_{k-1}^{(i)} P\left(z_k | X_{0:k}^{(i)}, Z_{0:k-1}\right) \tag{34}$$

For FastSLAM 2.0, the proposal distribution includes the current observation

$$x_k^{(i)} \sim P\left(x_k | X_{0:k-1}, Z_{0:k}, u_k\right), \tag{35}$$

where

$$P\left(x_k | X_{0:k-1}, Z_{0:k}, u_k\right) = \frac{1}{C} P\left(z_k | x_k, X_{0:k-1}^{(i)}, Z_{0:k-1}\right) P\left(x_k | x_{k-1}^{(i)}, u_k\right),$$

and C is a normalising constant. The importance weight according to (31) is $w_k^{(i)} = w_{k-1}^{(i)} C$. The advantage of FastSLAM 2.0 is that its proposal distribution is locally optimal[81]. That is, for each particle, it gives the smallest possible variance in importance weight $w_k^{(i)}$ conditioned upon the available information, $X_{0:k-1}^{(i)}$, $Z_{0:k}$ and $U_{0:k}$.

Maurelli *et al.*[82] presented a particle filter approach for localisation of autonomous underwater vehicles both in structured and unstructured environments. Experimental results show the high performances of this algorithm, which is robust to noisy measurements. Fairfield *et al.* [83] described the application of a Rao-Blackwellized Particle Filter to the problem of simultaneous localisation and mapping on-board a hovering autonomous underwater vehicle, which is called DEPTHX. Due to the three dimensional nature of the tunnels, the Particle Filter must use a three dimensional map. The SLAM algorithm used 20 particles with 2003 voxel maps at 1m resolution (see Figure 7 for the final map of a single particle). Each run took about 43 minutes on a modern desktop computer: about four times faster than realtime. The SLAM error is higher than localisation only, but much lower than dead-reckoning. More importantly, the SLAM error is bounded.

4.4.4 Data Association

Data association is the process of matching observations that are received by the filter to the features to which they correspond[84]. Data association is essential to the operation of the SLAM. The estimated location of landmark positions rely on the accuracy of the vehicle location estimate. An incorrect association of an observation to the map can cause the filter to diverge from a consistent estimate, effectively rendering all future predicted observations incorrect.

The most common method for associating observations to features in the map relies on Nearest Neighbour (NN) techniques. Nearest Neighbour association is taken in this case to be the closest association in a statistical sense. A common statistical discriminator is based on the normalised innovation squared between two estimates. Given an observation $z(k)$ comprising a range and bearing to the observed landmark, the innovation, $\upsilon(k) = z(k) - \hat{z}^{-}(k)$, and innovation covariance, $S(k)$ (see equation (12)). The normalised innovation squared between the observation and the estimated feature location is then compared against a validation gate, d_{min}, for the association being considered.

$$d_{fi} = \upsilon^T(k) S^{-1}(k) \upsilon(k) < d_{min} \tag{36}$$

The normalised innovation squared forms a χ^2 distribution that can be used to accept or reject a particular association with a given confidence level by the appropriate selection of d_{min}[84].

The NN uses the gate validation test to initially determine which landmark states are valid candidates to update with the observation received. Among all the possible landmarks, the one that is nearest to the observation is selected. All the other possible hypotheses are ignored. The NN can fail to recover the true data association when validation gates overlap, and the observations fall within this overlapped region. In such cases, the NN can associate the observation with an incorrect state. Such false data associations are known to induce catastrophic failures to the SLAM problem[85]. This is one flaw of NN. To overcome the low reliability of the simple NN and improve the robustness of data association in SLAM, various work has examined the possibility of using Multiple Hypothesis Tracking (MHT) implementation[86], joint compatibility test based on the branch and bound search (JCBB) [87].

4.4.5 Methods for improving the robustness of SLAM

It is an important research issue to figure out methods to improve the robustness of SLAM so that it can adapt itself to the complex underwater environment. Among all these methods, data association is a significant technical tool to enhance the robustness of SLAM algorithms. The developing trend of AUV navigation leads to the expansion of the map and the increase of the computational complexity, which have been inhibiting the application of SLAM algorithm for a long time. In addition, the underwater scenario is extremely complex, the sensors that can be applied are limited to sonar, underwater camera, underwater acoustic positioning system etc. Furthermore, the obtained observation has large noise and interference, which demands the high performance of the data association. Therefore, to solve these problems satisfactorily, researchers in literature have done plenty of work.

RibasD [88] proposed Individual Compatibility Nearest Neighbor (ICNN), the Individual Compatibility (IC) between an observation and a feature in the map can be determined using an innovation test that measures the Mahalanobis distance[89]. Then, from the features which satisfy the IC criterion, the one with the smallest distance is chosen. In spite of its conceptual simplicity and computational efficiency, the ICNN test has low reliability in detecting spurious matchings when the vehicle error grows with respect to sensor error. This happens mainly because it only considers individual compatibility between an observation and a feature, but ignores the fact that this pairing must be jointly compatible with other individual compatible pairings to form a consistent hypothesis.

A much more restrictive algorithm that considers this aspect is the Joint Compatibility Branch and Bound (JCBB)[87]. This approach explores the space of correspondence using a Branch and Bound (BB) search algorithm to find the hypothesis containing the largest number of Jointly Compatible (JC) pairings. Leonard *et al.*[90] verified the two data association methods ICNN and JCBB, provided the simulation result, put forward several problems needed to be solved. williams *et al.* [84] proposed a novel feature initialisation scheme designed to improve the performance of the SLAM algorithm. Rather than discarding observations of tentative features in the environment, this initialisation scheme incorporates them into the filter. When a feature is confirmed through multiple sightings, the information is consolidated into a single estimate of the feature through the application of appropriately formulated constraints.

4.4.6 Methods for reducing the computational complexity

Computational complexity is one of the main issues of SLAM. Although the computational complexity can be reduced to be proportional to the square of the number of features, it is still far and away from the real-time requirement as future AUV will go deeper and further sea. Consequently, improving the computational efficiency is the key aspects of the development SLAM algorithms.

One of the means of reducing the computational complexity is called Extended Information Filter (EIF), which was proposed by Thrun *et al.* [91] in 2002. Instead of applying covariance matrix to represent the correlation of space information , it utilizes space information matrix to do that, and uses local, Web-like networks of features to maintain the map of the environment feature. The advantage of this is that the calculation in update stage has nothing to do with the environment features, keeping the fixed time cost, thus conquring the flaws of massive calculation. However, the theory and implementation is in its preliminary, there exists many problems to be tackled.

Another method of reducing the computational complexity is the sub-mapping method, which divides the global map into relatively smaller local sub map. Several questions should be taken into account when using this method, for instance, how to partition the global map, how to represent the relationship between sub maps, how to keep the consistency of the global map when transmitting information between the global map and the sub maps. So far, to the knowledge of the author, the sub mapping methods mainly include Decoupled Stochastic Mapping (DSM) proposed by Leonard *et al*. in [92], Constrained Local Submap Filter (CLSF) proposed by Williams *et al*. in [93] and Compressed Extended Kalman Filter (ECKF) [94].

In addition, some researchers conducted work on Cooperative Simultaneous Localisation and Mapping(CSLAM) to realise the reduction of the computational complexity[95, 96]. Compared with single vehicle, CSLAM can increase the mapping efficiency and the accuracy through the cooperation and the information sharing between multi vehicles. Walter *et al*. [97] proposed a SLAM algorithm for 3 land robots in the simulation of underwater environment with low visibility, adopting delayed Kalman filter. The experiment showed that the CSLAM improved the accuracy of navigation. Due to the limitation of bandwidth of the underwater communication, observation techniques, CSLAM is not applied to AUV at present, which needs to develop and research.

4.4.7 Other solutions

Paskin [98] proposed a Thin Junction Tree Filters for SLAM (TJTF), trying to apply dynamic Bayesian network (DBN) approximation inference to solve SLAM problem. The basic principle of TJTF is: Consider the states in Kalman filter as a time-variant Gaussian model, apply the dependency of direct or indirect nodes to represent the correlation, and then to find and discard the weak and redundant dependency thus to simplify the reasoning process by analysing the varying process of the graphic. In the SLAM domain, TJTF can be made to (frequently) operate in constant-time, making its time complexity competitive with that of the FastSLAM algorithm.

TJTF also represents a single estimate of the map, incontrast to FastSLAM. while this improves the complexity of data association and extracting map marginals,it significantly curbs the representations expressiveness. Finally, while FastSLAM relies on the diversity of the particle population to represent correlations between landmark estimates (and is therefore sensitive to noise due to the curse of dimensionality), TJTF chooses the optimal correlations to represent directly, subject to a computational complexity constraint; thus, it is less susceptible to divergence.

Jaulin [99] proposed a set membership method based on interval analysis to solve the SLAM problem. The principle of the approach is to cast the SLAM problem into a constraint satisfaction problem for which interval propagation algorithms are particularly powerful. The main advantages of the interval propagation approach are its generality, its simplicity, its reliability, and its efficiency. Nevertheless, the interval constraint propagation SLAM method is offline and requires a human operator to detect seamarks and make the mark associations.

There are also other new methods for SLAM, for example, A genetic algorithm for SLAM[100] and A multigrid approach for accelerating relaxation-based SLAM[101].

4.5 Application of SLAM for AUVs

Underwater SLAM has many more challenging issues compared to indoor and outdoor SLAM, due to the unstructured nature of the underwater scenarios and the difficulty to identify reliable features. Many underwater features are scale dependant, sensitive to viewing angle and scale. To tackle these challenging issues, many researchers in the world conduct a plenty of research work on SLAM for AUVs. Regarding the process of feature extraction, much work focused on using point feature. For instance, Williams *et al.*[102] used point feature to build the map, this approach proposed to fuse information from the vehicle's on-board sonar and vision systems. They use EKF based SLAM combined with sonar and vision to obtain 3D structure and texture.

He *et al.*[103] from Ocean University of China described a localisation and mapping system for autonomous underwater vehicles (AUV) which used a DVL (Doppler velocity log) sensor and AHRS (attitude and heading reference system) sensor to measure AUV's depth, attitude and velocities relative to the bottom. A mechanically scanning imaging sonar (MSIS) is employed to obtain acoustic images of objects in underwater environment. In order to estimate optimally AUV pose without *a priori* map of the environment, simultaneous localisation and map building (SLAM), a prevailing method in the past decade, is presented based on point features extraction and EKF-based estimator. Using Fluvia Nautic marina data set, the proposed method was compared with traditional dead-reckoning, results show that their solution can reduce estimation error significantly. Leonard *et al.*[92] and Bosse *et al.* [104] also used point features. The former implemented the decoupled stochastic mapping and performed tests on a water tank, while the later proposed the constant time SLAM and used LBL information to help on the localisation.

Majumder *et al.* from Australian Centre for Field Robotics at the University of Sydney [105] developed an AUV Oberon to conduct underwater experiment in the offshore of Sydney, the AUV was equipped with cone-shaped sensor of 585 kHz, fan-shaped scanning sensor, underwater colour camera, fibre-optical gyroscope and pressure sensor. They fuse the information of sonar and image captured from the camera. The experiment proves that the estimation generated from SLAM algorithm is continuously consistent.

Ribas *et al.*[106] in University of Girona described a system for underwater navigation with AUVs in partially structured environments, such as dams, ports or marine platforms. An imaging sonar is used to obtain information about the location of planar structures present in such environments. This information is incorporated into a feature-based SLAM algorithm in a two step process: (1) the full 360 sonar scan is undistorted (to compensate for vehicle motion), thresholded and segmented to determine which measurements correspond to planar environment features and which should be ignored; and (2) SLAM proceeds once the data association is obtained: both the vehicle motion and the measurements whose correct association has been previously determined are incorporated in the SLAM algorithm. This two step delayed SLAM process allows to robustly determine the feature and vehicle locations in the presence of large amounts of spurious or unrelated measurements that might correspond to boats, rocks, etc. Preliminary experiments show the viability of the proposed approach.

Ruiz *et al.* [107] from Heriot-Watt University in Edinburgh UK described a Concurrent Mapping and Localisation (CML) algorithm for localising an AUV, mainly focused on feature extraction and data association problem. The chosen targets consist of returns of a significant strength. The segmentation detects these targets and calculates (a) the relative position of their centre of mass with respect to the vehicle, (b) the targets surface size, and (c) the targets first invariant moment. This information is used by the system to perform the data association. The well known Multiple Hypothesis Tracking Filter (MHTF) data association algorithm associates the segmented targets to existing targets on the stochastic map. Segmented targets that are not associated to any of the tracked targets are used to initialise new map targets.

Aulinas *et al.*[108], from Institute of Informatics and Applications Girona/Spain, developed an AUV called SPARUS, using Inertial Measurement Unit (IMU) and the Doppler Velocity Log (DVL) to measure navigation data, while a down looking camera is used to gather data from the environment. They built a SLAM approach, called the selective submapping SLAM, using navigation readings to improve vehicle localisation, and the map through its correlation with the vehicle position. The main idea of this approach is to use EKF based SLAM to build local maps (x_i, P_i), where x_i is the state vector describing vehicles pose, vehicles velocities and the map, while P_i is its associated uncertainty. The size of these local maps is bounded by the total number of features and by the level of uncertainty. The relative topological relationship between consecutive local maps is stored in a global level map (x_G, P_G). The global level is used to search for loop closure (H_{Loop}), i.e. the vehicle is revisiting a region. The loop closing strategy involves a decision on whether to fuse local maps depending on the amount of found correspondences between submaps.

Underwater SLAM implementations have some points in common, for instance, imaging sonar is widely used, the most common filtering technique is the Extended Kalman Filter (EKF) and point features are commonly used to represent the map. Some approaches use side-scan sonar or optical cameras, which seems to become more important as technology advances. The use of EKF based SLAM has been shown to handle uncertainties properly; however, the computational cost associated with EKF grows with the size of the map. In addition, linearization errors accumulate in long missions, increasing the chance of producing inconsistent mapping solutions.

5 Conclusion

This report has provided an overview of relevant elements of localisation and mapping for Autonomous Underwater Vehicles (AUV). First, an brief introduction of the concept and the historical development of AUV is presented; then a relatively detailed description of sensors is provided; finally, a comprehensive investigation of the solutions to the SLAM problem for AUVs are conducted.

From what has been discussed in this report, it is clear that a great deal of research work has been conducted to realize the localization and navigation for AUVs. Although there are some successful examples, various challenging issues remain and require researchers to pay more attention, some of which are:

1. In underwater navigation, the dynamic and unstructured characteristics of underwater environments require sensors with a high resolution and accuracy. This is very challenge.

2. In the situation that the environmental feature is not intuitive, it is necessary to apply proactive SLAM such as deploying artificial landmarks to explore useful information proactively. However, this will increase the cost and complexity of the AUV navigation system.

3. Since high accurate sensor systems such as LBL, USBL and SBL have a large size and high cost, it is impractical to use these sensor systems for localisation of small bio-inspired vehicles such as robotic fish. Consequently, it is highly desirable to conduct research on improving the accuracy of SLAM for the small AUVs.

In the future, in spite of the difficulties existing in realizing highly accurate SLAM for AUVs, we believe more and more accurate and robust localization solutions will be achieved with the development of both sensors and SLAM algorithms.

References

[1] [Online]. Available: http://en.wikipedia.org/wiki/Autonomous_underwater_vehicle

[2] J. Yuh, "Design and control of autonomous underwater robots: A survey," *Autonomous Robots*, vol. 8, no. 1, pp. 7–24, 2000.

[3] R. Wernli, "Auvs-the maturity of the technology," in *OCEANS'99 MTS/IEEE. Riding the Crest into the 21st Century*, vol. 1. IEEE, 1999, pp. 189–195.

[4] D. Blidberg, "The development of autonomous underwater vehicles (auvs); a brief summary," in *IEEE ICRA*. Citeseer, 2001.

[5] H. Hu, "Biologically inspired design of autonomous robotic fish at essex," in *IEEE SMC Chapter Conference on Advances in Cybernetic Systems*. Citeseer, 2006.

[6] J. Yu, M. Tan, S. Wang, and E. Chen, "Development of a biomimetic robotic fish and its control algorithm," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 34, no. 4, pp. 1798–1810, 2004.

[7] J. Leonard, A. Bennett, C. Smith, and H. Feder, "Autonomous underwater vehicle navigation," in *IEEE ICRA Workshop on Navigation of Outdoor Autonomous Vehicles*. Citeseer, 1998.

[8] [Online]. Available: http://en.wikipedia.org/wiki/GPS

[9] [Online]. Available: http://en.wikipedia.org/wiki/Inertial_navigation_system

[10] R. Gray and P. Maybeck, "An integrated gps/ins/baro and radar altimeter system for aircraft precision approach landings," in *Aerospace and Electronics Conference, 1995. NAECON 1995., Proceedings of the IEEE 1995 National*, vol. 1. IEEE, 1995, pp. 161–168.

[11] K. Braden, C. Browning, H. Gelderloos, F. Smith, C. Marttila, and L. Vallot, "Integrated inertial navigation system/global positioning system (ins/gps) for manned return vehicle autoland application," in *Position Location and Navigation Symposium, 1990. Record. The 1990's-A Decade of Excellence in the Navigation Sciences. IEEE PLANS'90., IEEE*. IEEE, 1990, pp. 74–82.

[12] G. Barnes, S. Lee, and A. Perez, "Ins/gps space application for all flight dynamics-boost through landing," in *Position Location and Navigation Symposium, 1996., IEEE 1996*. IEEE, 1996, pp. 457–463.

[13] C. Jekeli, "Balloon gravimetry using gps and ins," *Aerospace and Electronic Systems Magazine, IEEE*, vol. 7, no. 6, pp. 9–15, 1992.

[14] R. Ornedo, K. Farnsworth, and G. Sandhoo, "Gps and radar aided inertial navigation system for missile system applications," in *Position Location and Navigation Symposium, IEEE 1998*. IEEE, 1998, pp. 614–621.

[15] M. Martin and D. Vause, "New low cost avionics with ins/gps for a variety of vehicles," *Aerospace and Electronic Systems Magazine, IEEE*, vol. 13, no. 11, pp. 41–46, 1998.

[16] R. Tranfield, "Ins/gps navigation systems for land applications," in *Position Location and Navigation Symposium, 1996., IEEE 1996*. IEEE, 1996, pp. 391–398.

[17] B. Barshan and H. Durrant-Whyte, "Inertial navigation systems for mobile robots," *Robotics and Automation, IEEE Transactions on*, vol. 11, no. 3, pp. 328–342, 1995.

[18] S. Sukkarieh, E. Nebot, and H. Durrant-Whyte, "Achieving integrity in an ins/gps navigation loop for autonomous land vehicle applications," in *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, vol. 4. IEEE, 1998, pp. 3437–3442.

[19] X. Yun, E. Bachmann, R. McGhee, R. Whalen, R. Roberts, R. Knapp, A. Healey, and M. Zyda, "Testing and evaluation of an integrated gps/ins system for small auv navigation," *Oceanic Engineering, IEEE Journal of*, vol. 24, no. 3, pp. 396–404, 1999.

[20] D. Loebis, R. Sutton, J. Chudley, and W. Naeem, "Adaptive tuning of a kalman filter via fuzzy logic for an intelligent auv navigation system," *Control engineering practice*, vol. 12, no. 12, pp. 1531–1539, 2004.

[21] L. Zhao and W. Gao, "The experimental study on gps/ins/dvl integration for auv," in *Position Location and Navigation Symposium, 2004. PLANS 2004.* IEEE, 2004, pp. 337–340.

[22] [Online]. Available: http://en.wikipedia.org/wiki/Side-scan_sonar

[23] Y. Petillot, S. Reed, and J. Bell, "Real time auv pipeline detection and tracking using side scan sonar and multi-beam echo-sounder," in *OCEANS'02 MTS/IEEE*, vol. 1. IEEE, 2002, pp. 217–222.

[24] S. Reed, Y. Petillot, and J. Bell, "An automatic approach to the detection and extraction of mine features in sidescan sonar," *Oceanic Engineering, IEEE Journal of*, vol. 28, no. 1, pp. 90–105, 2003.

[25] H. Johnson and M. Helferty, "The geological interpretation of side-scan sonar," *Reviews of Geophysics*, vol. 28, no. 4, pp. 357–380, 1990.

[26] D. Langer and M. Hebert, "Building qualitative elevation maps from side scan sonar data for autonomous underwater navigation," in *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on.* IEEE, 1991, pp. 2478–2483.

[27] I. Ruiz, Y. Petillot, and D. Lane, "Improved auv navigation using side-scan sonar," in *OCEANS-CONFERENCE-*, vol. 3. IEEE; 1998, 2003, pp. 1261–1268.

[28] I. Tena Ruiz, S. de Raucourt, Y. Petillot, and D. Lane, "Concurrent mapping and localization using sidescan sonar," *Oceanic Engineering, IEEE Journal of*, vol. 29, no. 2, pp. 442–456, 2004.

[29] J. Salvi, Y. Petillot, S. Thomas, and J. Aulinas, "Visual slam for underwater vehicles using video velocity log and natural landmarks," in *OCEANS 2008.* IEEE, 2008, pp. 1–6.

[30] P. Zhang, E. Milios, and J. Gu, "Underwater robot localization using artificial visual landmarks," in *Robotics and Biomimetics, 2004. ROBIO 2004. IEEE International Conference on.* IEEE, 2004, pp. 705–710.

[31] M. Carreras, P. Ridao, R. Garcia, and T. Nicosevici, "Vision-based localization of an underwater robot in a structured environment," in *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, vol. 1. IEEE, 2003, pp. 971–976.

[32] G. Karras, D. Panagou, and K. Kyriakopoulos, "Target-referenced localization of an underwater vehicle using a laser-based vision system," in *OCEANS 2006.* IEEE, 2006, pp. 1–6.

[33] [Online]. Available: http://en.wikipedia.org/wiki/Underwater_Acoustic_Positioning_System#cite_note-3

[34] B. Foley and D. Mindell, "Precision survey and archaeological methodology in deep water," *ENALIA: The Journal of the Hellenic Institute of Marine Archaeology, VI*, pp. 49–56, 2002.

[35] A. Matos, N. Cruz, A. Martins, and F. Lobo Pereira, "Development and implementation of a low-cost lbl navigation system for an auv," in *OCEANS'99 MTS/IEEE. Riding the Crest into the 21st Century*, vol. 2. IEEE, 1999, pp. 774–779.

[36] P. Rigby, O. Pizarro, and S. Williams, "Towards geo-referenced auv navigation through fusion of usbl and dvl measurements," in *OCEANS 2006*. IEEE, 2006, pp. 1–6.

[37] S. LI, C. TAO, and G. BAO, "Ins/usbl underwater navigation system based on kalman filter [j]," *Ocean Technology*, vol. 3, 2008.

[38] M. Morgado, P. Oliveira, C. Silvestre, and J. Vasconcelos, "Usbl/ins tightly-coupled integration technique for underwater vehicles," in *Proceedings Of The 9th International Conference on Information Fusion*. Citeseer, 2006.

[39] A. Alcocer, P. Oliveira, and A. Pascoal, "Underwater acoustic positioning systems based on buoys with gps," in *Proceedings of the Eighth European Conference on Underwater Acoustics*. Citeseer, 2006, pp. 12–15.

[40] ——, "Study and implementation of an ekf gib-based underwater positioning system," *Control engineering practice*, vol. 15, no. 6, pp. 689–701, 2007.

[41] A. Elfes, "Sonar-based real-world mapping and navigation," *Robotics and Automation, IEEE Journal of*, vol. 3, no. 3, pp. 249–265, 1987.

[42] ——, "Occupancy grids: A probabilistic framework for robot perception and navigation," Ph.D. dissertation, Carnegie Mellon University., 1989.

[43] H. Moravec, "Sensor fusion in certainty grids for mobile robots," *AI magazine*, vol. 9, no. 2, p. 61, 1988.

[44] J. Borenstein and Y. Koren, "The vector field histogram-fast obstacle avoidance for mobile robots," *Robotics and Automation, IEEE Transactions on*, vol. 7, no. 3, pp. 278–288, 1991.

[45] S. Thrun, "Exploration and model building in mobile robot domains," in *Neural Networks, 1993., IEEE International Conference on*. IEEE, 1993, pp. 175–180.

[46] J. van Dam, B. Kr
"ose, and F. Groen, "Neural network applications in sensor fusion for an autonomous mobile robot," *Reasoning with Uncertainty in Robotics*, pp. 263–278, 1996.

[47] H. Moravec and A. Elfes, "High resolution maps from wide angle sonar," in *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*, vol. 2. IEEE, 1985, pp. 116–121.

[48] D. Cho, "Certainty grid representation for robot navigation by a bayesian method." *Robotica*, vol. 8, no. 2, pp. 159–165, 1990.

[49] G. Oriolo, M. Vendittelli, and G. Ulivi, "On-line map building and navigation for autonomous mobile robots," in *Robotics and Automation, 1995. Proceedings., 1995 IEEE International Conference on*, vol. 3. IEEE, 1995, pp. 2900–2906.

[50] E. Fabrizi and A. Saffiotti, "Extracting topology-based maps from gridmaps," in *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, vol. 3. IEEE, 2000, pp. 2972–2978.

[51] S. Thrun and A. B
"ucken, "Integrating grid-based and topological maps for mobile robot navigation," in *Proceedings of the National Conference on Artificial Intelligence.* Citeseer, 1996, pp. 944–951.

[52] S. Pfister, S. Roumeliotis, and J. Burdick, "Weighted line fitting algorithms for mobile robot map building and efficient data representation," in *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, vol. 1. IEEE, 2003, pp. 1304–1311.

[53] J. Crowley, "Dynamic world modelling for an intelligent mobile robot using a rotating ultra-sonic sensing device," *IEEE International Conference on Roboitcs and Automation (ICRA '85)'*, vol. 1, pp. 128–135, 1985.

[54] R. Chatila, "Mobile robot navigation: Space modeling and decisional processes," in *Robotics Research: The Third International Symposium, 0. Faugeras and G. Giralt (Eds), MIT Press, Cambridge, Massachusetts*, 1986.

[55] R. Kuc and M. Siegel, "Physically based simulation model for acoustic sensor robot navigation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, no. 6, pp. 766–778, 1987.

[56] U. Larsson, J. Forsberg, and A. Wernersson, "On robot navigation using identical landmarks: integrating measurements from a time-of-flight laser," in *Multisensor Fusion and Integration for Intelligent Systems, 1994. IEEE International Conference on MFI'94.* IEEE, 1994, pp. 17–26.

[57] P. Lamon, I. Nourbakhsh, B. Jensen, and R. Siegwart, "Deriving and matching image fingerprint sequences for mobile robot localization," in *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, vol. 2. IEEE, 2001, pp. 1609–1614.

[58] H. Durrant-Whyte, "Sensor models and multisensor integration," *The International Journal of Robotics Research*, vol. 7, no. 6, p. 97, 1988.

[59] S. Engelson, "Passive map learning and visual place recognition," Ph.D. dissertation, Yale University, 1994.

[60] R. Brooks, "Visual map making for a mobile robot," in *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*, vol. 2. IEEE, 1985, pp. 824–829.

[61] D. Kortenkamp and T. Weymouth, "Topological mapping for mobile robots using a combination of sonar and vision sensing," in *Proceedings of the National Conference on Artificial Intelligence.* JOHN WILEY & SONS LTD, 1995, pp. 979–979.

[62] B. Kuipers and Y. Byun, "A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations," *Robotics and autonomous systems*, vol. 8, no. 1-2, pp. 47–63, 1991.

[63] D. Pierce and B. Kuipers, "Learning to explore and build maps," in *Proceedings of the National Conference on Artificial Intelligence.* JOHN WILEY & SONS LTD, 1995, pp. 1264–1264.

[64] S. Thrun, "Learning metric-topological maps for indoor mobile robot navigation* 1," *Artificial Intelligence*, vol. 99, no. 1, pp. 21–71, 1998.

[65] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part i," *Robotics & Automation Magazine, IEEE*, vol. 13, no. 2, pp. 99–110, 2006.

[66] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," *Autonomous robot vehicles*, vol. 1, pp. 167–193, 1990.

[67] P. Moutarlier and R. Chatila, "Stochastic multisensory data fusion for mobile robot location and environment modeling," in *5th Int. Symposium on Robotics Research*, vol. 1, 1989.

[68] J. Leonard, H. Durrant-Whyte, and I. Cox, "Dynamic map building for an autonomous mobile robot," *The International Journal of Robotics Research*, vol. 11, no. 4, p. 286, 1992.

[69] H. Durrant-Whyte, D. Rye, and E. Nebot, "Localisation of automatic guided vehicles," in *Robotics Research: The 7th International Symposium (ISRR95)*. New York: Springer Verlag, 1996, pp. 613–625.

[70] M.Csorba, "Simultaneous localization and map building," Ph.D. dissertation, Univ. Oxford, 1997.

[71] D. Ribas, P. Ridao, J. Tardós, and J. Neira, "Underwater slam in man-made structured environments," *Journal of Field Robotics*, vol. 25, no. 11-12, pp. 898–921, 2008.

[72] I. Mahon and S. Williams, "Slam using natural features in an underwater environment," in *Control, Automation, Robotics and Vision Conference, 2004. ICARCV 2004 8th*, vol. 3. IEEE, 2004, pp. 2076–2081.

[73] [Online]. Available: http://en.wikipedia.org/wiki/Extended_Kalman_filter

[74] E. Wan and R. Van Der Merwe, "The unscented kalman filter," *Kalman filtering and neural networks*, pp. 221–280, 2001.

[75] S. Julier and J. Uhlmann, "A new extension of the kalman filter to nonlinear systems," in *Int. Symp. Aerospace/Defense Sensing, Simul. and Controls*, vol. 3. Citeseer, 1997, p. 26.

[76] J. Lian, W. Wu, T. Li, and X. Hu, "Application of unscented kalman filter in auv integrated navigation," *Journal of Chinese Inertial Technology*, vol. 1, 2005.

[77] M. LIU, J. HU, and W. LI, "Research on integrated navigation for autonomous underwater vehicle based on an improved unscented kalman filter," *Acta Armamentarii*, p. 02, 2011.

[78] S. Thrun, M. Montemerlo, D. Koller, B. Wegbreit, J. Nieto, and E. Nebot, "Fastslam: An efficient solution to the simultaneous localization and mapping problem with unknown data association," *Journal of Machine Learning Research*, vol. 4, no. 3, pp. 380–407, 2004.

[79] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "Fastslam: A factored solution to the simultaneous localization and mapping problem," in *Proceedings of the National conference on Artificial Intelligence*. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2002, pp. 593–598.

[80] ——, "Fastslam 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges," in *International Joint Conference on Artificial Intelligence*, vol. 18. Citeseer, 2003, pp. 1151–1156.

[81] A. Doucet, S. Godsill, and C. Andrieu, *On sequential simulation-based methods for Bayesian filtering*. Univ., Dep. of Engineering, 1998.

[82] F. Maurelli, S. Krupinski, Y. Petillot, and J. Salvi, "A particle filter approach for auv localization," in *OCEANS 2008*. IEEE, 2008, pp. 1–7.

[83] N. Fairfield, G. Kantor, and D. Wettergreen, "Towards particle filter slam with three dimensional evidence grids in a flooded subterranean environment," in *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*. IEEE, 2006, pp. 3575–3580.

[84] S. Williams, H. Durrant-Whyte, and G. Dissanayake, "Constrained initialization of the simultaneous localization and mapping algorithm," *The International Journal of Robotics Research*, vol. 22, no. 7-8, p. 541, 2003.

[85] J. Nieto, J. Guivant, E. Nebot, and S. Thrun, "Real time data association for fastslam," in *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, vol. 1. IEEE, 2003, pp. 412–418.

[86] C. Smith, H. Feder, and J. Leonard, "Multiple target tracking with navigation uncertainty," in *Decision and Control, 1998. Proceedings of the 37th IEEE Conference on*, vol. 1. IEEE, 1998, pp. 760–761.

[87] J. Neira and J. Tardós, "Data association in stochastic mapping using the joint compatibility test," *Robotics and Automation, IEEE Transactions on*, vol. 17, no. 6, pp. 890–897, 2001.

[88] RibasD, "Towards simultaneous localization and mapping for an auv using an imaging sonar," Department of electronics, informatics and automation, Universitat de Girona, Tech. Rep., 2005.

[89] [Online]. Available: http://en.wikipedia.org/wiki/Mahalanobis_distance

[90] J. Leonard, P. Newman, R. Rikoski, J. Neira, and J. Tardós, "Towards robust data association and feature modeling for concurrent mapping and localization," *Robotics Research*, pp. 7–20, 2003.

[91] S. Thrun, D. Koller, Z. Ghahramani, H. Durrant-Whyte, and A. Ng, "Simultaneous mapping and localization with sparse extended information filters: Theory and initial results," *Algorithmic Foundations of Robotics V*, pp. 363–380, 2004.

[92] J. Leonard and H. Feder, "Decoupled stochastic mapping [for mobile robot & auv navigation]," *Oceanic Engineering, IEEE Journal of*, vol. 26, no. 4, pp. 561–571, 2001.

[93] S. Williams, "Efficient solutions to autonomous mapping and navigation problems," Ph.D. dissertation, Citeseer, 2001.

[94] J. Guivant and E. Nebot, "Optimization of the simultaneous localization and map-building algorithm for real-time implementation," *Robotics and Automation, IEEE Transactions on*, vol. 17, no. 3, pp. 242–257, 2001.

[95] M. Di Marco, A. Garulli, A. Giannitrapani, and A. Vicino, "Simultaneous localization and map building for a team of cooperating robots: a set membership approach," *Robotics and Automation, IEEE Transactions on*, vol. 19, no. 2, pp. 238–249, 2003.

[96] J. Fenwick, P. Newman, and J. Leonard, "Cooperative concurrent mapping and localization," in *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, vol. 2. IEEE, 2002, pp. 1810–1817.

[97] M. Walter and J. Leonard, "An experimental investigation of cooperative slam," in *Proceedings of the Fifth IFAC/EURON Symposium on Intelligent Autonomous Vehicles,(Lisbon, Portugal)*, 2004.

[98] M. Paskin, "Thin junction tree filters for simultaneous localization and mapping," *Computer*, 2002.

[99] L. Jaulin, "A nonlinear set membership approach for the localization and map building of underwater robots," *Robotics, IEEE Transactions on*, vol. 25, no. 1, pp. 88–98, 2009.

[100] T. Duckett, "A genetic algorithm for simultaneous localization and mapping," in *IEEE International Conference on Robotics and Automation*, vol. 1. Citeseer, 2003, pp. 434–439.

[101] U. Frese and T. Duckett, "A multigrid approach for accelerating relaxation-based slam," *Autonome Mobile Systeme 2003: 18. Fachgespr*
"ach Karlsruhe, 4./5. Dezember 2003, vol. 18, p. 192, 2003.

[102] S. Williams and I. Mahon, "Simultaneous localisation and mapping on the great barrier reef," in *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, vol. 2. IEEE, 2004, pp. 1771–1776.

[103] B. He, K. Yang, S. Zhao, and Y. Wang, "Underwater simultaneous localization and mapping based on ekf and point features," in *Mechatronics and Automation, 2009. ICMA 2009. International Conference on*. IEEE, 2009, pp. 4845–4850.

[104] M. Bosse, P. Newman, J. Leonard, and S. Teller, "Simultaneous localization and map building in large-scale cyclic environments using the atlas framework," *The International Journal of Robotics Research*, vol. 23, no. 12, p. 1113, 2004.

[105] S. Majumder, S. Scheding, and H. Durrant-Whyte, "Multisensor data fusion for underwater navigation," *Robotics and Autonomous Systems*, vol. 35, no. 2, pp. 97–108, 2001.

[106] D. Ribas, P. Ridao, J. Neira, and J. Tardós, "Slam using an imaging sonar for partially structured underwater environments," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*. IEEE, 2006, pp. 5040–5045.

[107] I. Tena Ruiz, Y. Petillot, D. Lane, and C. Salson, "Feature extraction and data association for auv concurrent mapping and localisation," in *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, vol. 3. IEEE, 2001, pp. 2785–2790.

[108] J. Aulinas, Y. Petillot, X. Lladó, J. Salvi, and R. Garcia, "Vision-based underwater slam for the sparus auv."