

Application of Forward-looking Scanning Sonar on Simultaneous Localization and Mapping for an Autonomous Underwater Vehicle

Shu Tian, Bo He⁺, Shujuan Chen and Guijie Liu

College of Information Science and Engineering, Ocean University of China

Qingdao, China

Abstract—Simultaneous localization and mapping (SLAM) is one of the most promising ways for an autonomous underwater vehicle (AUV) to achieve autonomous navigation. Sonar, for the advantage of sound transmission in water, plays an important role in acquiring underwater environmental information to implement SLAM. In this paper, we will introduces an AUV system with a forward-looking scanning sonar, describing how to process sonar data to extract environmental features and furtherly apply them to SLAM.

Keywords-Forward-looking scanning sonar; SLAM; AUV

1. Introduction

In recent years, autonomous underwater vehicle (AUV) is becoming a hot topic in the ocean technology. To realize the autonomous navigation [1], simultaneous localization and mapping (SLAM) or called concurrent mapping and localization (CML) is mainly used. That's, in an unknown environment, without knowing the location of itself, the vehicle incrementally builds a map of the surroundings by extracting information from all kinds of sensors, and simultaneously uses the map to compute its location. One of the most important sensors used in SLAM is sonar, which has an advantage over laser or infrared ray for the low attenuation of sound propagating in water. In this paper, we use a forward-looking scanning sonar to extract the point features of the underwater environment and with the information from some other sensors, SLAM is finally achieved.

The organization of this paper is as follows: part 2 gives a brief introduction to our AUV platform, part 3 will describe the method of processing sonar data and the SLAM algorithm we use is explained in part 4. Part 5 will show results of our experiment and also give a conclusion to the whole work.

2. Auv Platform

Figure 1 (left) shows the picture of AUV platform we use. This AUV, named "C-Ranger", is mainly made up of the central computer unit, data acquisition unit, power supply unit and several onboard sensors, including GPS (Global Position System), DVL (Doppler Velocity Log), AHRS (Attitude Heading Reference System), a digital compass, a gyroscope and a forward-looking scanning sonar. Our vehicle now can achieve the aim of autonomous navigation.



⁺ Corresponding author.
E-mail address: bhe@ouc.edu.cn

Fig. 1: AUV platform (left) and sonar (right).

GPS, with a tolerance of less than 10m, gives a accurate position of the vehicle. The position information from GPS is used as a standard when evaluating the SLAM results. DVL offers the vehicle’s velocity of three directions. The heading and acceleration of the vehicle are measured by AHRS. Digital compass can detect directions accurately and the gyroscope is used to measure angular velocity.

Figure 1 (right) shows the sonar we use in the experiment. It’s a dual frequency CHIRP sonar from the Tritech company [2]. The standard maximum operational depth it can reach is 4000m and its maximum scanning range is up to 300m. Range resolution is variable from 5mm to 400mm, depending on the range.

By rotating, the sonar sends a sound beam at each step. When the beam meets an obstacle, echoes of different intensities are received. By collecting and processing these echo information, map of the surroundings is built. According to the different environments, we can change the range of beam and the step of rotation. Details of sonar data processing will be explained in part 3.

3. Sonar Data Processing

3.1. Sonar introduction

The sonar used here is a forward-looking scanning sonar. After setting the scanning range and rotating step, it sends a sound beam at each step and we call this beam a “ping”. The ping is divided into several segments, each segment is called a “bin”. The bins contain information about distance, bearing and intensity. When echoes of the bins are received, these information are used to draw a 2D sonar image that can represent the map of surroundings.

Sonar working principle and the format of sonar ping are showed in figure 2 [2]. As can be seen from figure 2 (up), the angle range can be set from 0° to 360°. In this paper, we choose 180° with a step of 0.9°. The range (also the length of the ping) is set to 100m. In figure 2 (down), each ping has 645 bytes. The first 44 bytes are parameters that control sonar’s frequency, range, angle, step, precision, etc. The following 600 bytes, marked in yellow, stand for the intensity of bins. The last byte is set to “0A” to be the end mark of a ping.

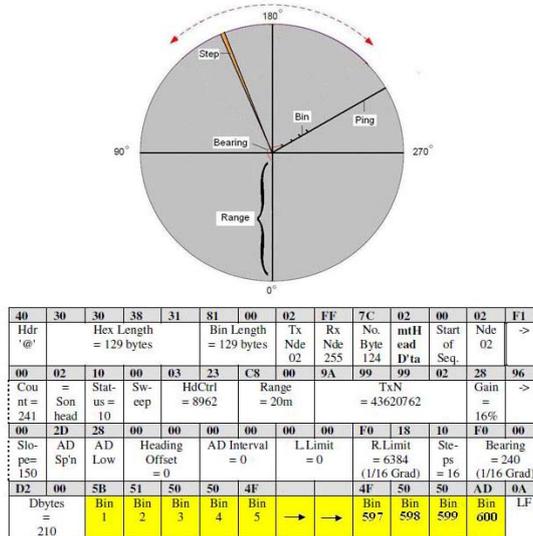


Fig. 2: Sonar working principle (up) and sonar ping format (down).

3.2. Data processing

As mentioned above, the 600 bytes bin data are used to draw a sonar image. The formulas to calculate the coordinates of a bin are as follows:

$$x(i) = Range / Nbins * i * \cos(\pi + Bearing) \tag{1}$$

$$y(i) = Range / Nbins * i * \sin(\pi + Bearing) \tag{2}$$

where *Range* is the distance sonar scans; *NBins* is the number of bins a ping is divided into; *i*, ranging from 1 to 600, is the sequence number of the bin; *Bearing* is the angle the ping is heading for.

Here, there's a problem of coordinate conversion. Figure 3 is the corresponding relationship between sonar coordinate and global coordinate. In this paper, all the data are transferred to the global coordinate.

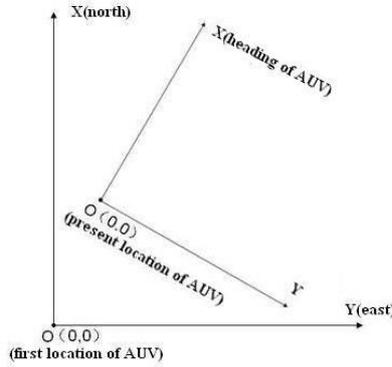


Fig. 3: Relationship between sonar coordinate and global coordinate.

Three steps are taken to process the sonar data:

1) Threshold segmentation. In this step, we define an intensity threshold, bins of lower intensity than the threshold are removed. In underwater environment, most of the low bin echoes are noises and interferences. This step reduces the data amount greatly.

2) Self-noise removal. Our C-Ranger AUV is about 2m long and since the sonar is bundled to the AUV, thus there's a 2m-radius noise circle around the sonar. Step 2 is to remove the noise of AUV itself.

3) Bin sparsing. When meeting an obstacle, echoes of bins are received by sonar. But the problem is different echoes may represent the same obstacle, that's to say, there's bin information redundancy. In this step, euclidean distance between two bins is calculated, if the distance is less than a certain value, then only one of them is kept. After this step, data amount is reduced while no useful information is lost, this remarkably improves the calculation speed. Formula of euclidean distance is:

$$d = \sqrt{(x_i - x_1)^2 + (y_i - y_1)^2} \quad (3)$$

3.3. Distortion compensation

After the above three steps, sonar image now can be drawn according to formula (1) and (2). Here, another problem is involved. Since step 2) and step 3) are implemented after each scanning period (that's 180° in our experiment), during this period, AUV is supposed to be static, which in fact, the vehicle is moving all the time with a speed of 0.5m/s to 1.5m/s. As a result, distortion caused by AUV's movement has to be solved.

In our research, we adopt two methods to solve this problem. One is using the AUV's position offered by dead reckoning algorithm, the other is to combine sonar data processing with SLAM algorithm, and by using SLAM's location of AUV to compensate the distortion. Both methods have received good results.

4. Slam Algorithm

The SLAM algorithm we use is based on EKF (Extended Kalman Filter). The basic idea of EKF-SLAM is to put AUV's attitude and environment features in one state vector, then during the movement, EKF is used to do the minimum mean square error estimation (MMSEE) with the observation of environment features, which will lead to autonomous navigation finally [3].

The state vector is:

$$X = [X_v \quad X_{f_1} \quad \dots \quad X_{f_n}]^T \quad (4)$$

Here, X_v is the attitude vector of AUV, $X_{f_i}(i=1,2,\dots,n)$ is the environment feature vector. The covariance matrix of X is:

$$P = \begin{bmatrix} P_v & P_{vf_1} & \cdots & P_{vf_n} \\ P_{f_1v} & P_{f_1} & \cdots & P_{f_1f_n} \\ \vdots & \vdots & \ddots & \vdots \\ P_{f_nv} & P_{f_nf_1} & \cdots & P_{f_n} \end{bmatrix} \quad (5)$$

where P_v , P_{vf_n} , P_{f_n} is the covariance matrix of AUV, between AUV and features, features respectively.

Environment features are observed as sonar scans, and EKF is used to estimate the state vector X and the covariance matrix P . The main steps of map building include prediction and update. Between the two steps, data association has to be implemented, and only those features which are relevant are used to update the attitude and the map [4].

4.1. Initialization

In our research, we adopt five variables to represent the AUV's attitude, including the coordinates (x_v, y_v) and velocities (v_{vx}, v_{vy}) of two direction, and heading angle (θ_v) the vehicle is moving for. So the initial state vector is:

$$X = X_v = [x_v \quad y_v \quad \theta_v \quad v_{vx} \quad v_{vy}]^T \quad (6)$$

The initial covariance matrix is:

$$P = P_v(0) = [0_{5 \times 5}] \quad (7)$$

4.2. Prediction

In this step, based on the last moment's attitude, the vehicle calculates its attitude at this moment. Since the features are unchanged during this process, so the prediction step only changes the vehicle's attitude vector X_v and the features vector X_{f_i} stays the same [4][5]:

$$X_v(k+1) = \begin{bmatrix} x_v(k) + x_\delta \cos \theta_v - y_\delta \sin \theta_v \\ y_v(k) + x_\delta \sin \theta_v + y_\delta \cos \theta_v \\ \theta_v(k) + \theta_\delta \\ v_{vx}(k) + aT \\ v_{vy}(k) + aT \end{bmatrix} \quad (8)$$

where x_δ and y_δ are the displacements between k and $k+1$ moment; θ_δ is the heading difference between this two moments; a is the acceleration and T is the prediction time interval.

The corresponding covariance matrix is:

$$P_{k+1} = GP_kG^T + HP_\delta H^T \quad (9)$$

where G , H are the Jacobian matrices with respect to the state X and the differentiation in the time interval. P_δ is the covariance matrix of the differentiation.

4.3. Sonar observation and data association

Assume that there's already a feature (x_i, y_i) in the map and the sonar observation value is [6]:

$$z = \begin{bmatrix} r \\ \theta \end{bmatrix} \quad (10)$$

$$R = \begin{bmatrix} \sigma_r^2 & \sigma_{r\theta}^2 \\ \sigma_{r\theta}^2 & \sigma_\theta^2 \end{bmatrix} \quad (11)$$

where r , θ are the distance and bearing between the vehicle and the feature; R is the observation covariance matrix. Transform the feature (x_i, y_i) to global coordinate:

$$\hat{z} = \begin{bmatrix} \sqrt{(x_i - x_v)^2 + (y_i - y_v)^2} \\ \arctan\left(\frac{y_i - y_v}{x_i - x_v}\right) - \theta \end{bmatrix} \quad (12)$$

If the observed z is correctly associated with (x_i, y_i) , then Kalman gain W is calculated as follows:

$$v = z - \hat{z} \quad (13)$$

$$S = HPH^T + R \quad (14)$$

$$W = PH^T S^{-1} \quad (15)$$

H is Jacobian matrix with respect to state vector X .

In data association part, we use the nearest neighbor (NN) algorithm [7]. Firstly, using formula (16) to calculate the Mahalanobis distance between the observed feature and the existed feature. If the Mahalanobis distance is less than the predetermined threshold, then the observed feature is considered to be associated with features that are already in the map and the update step is implemented. Else if the Mahalanobis distance is more than the threshold, which means that the observed feature should be added to the map as a new feature, the augment step is implemented.

$$D^2 = v^T S^{-1} v \quad (16)$$

1) Update. This step updates the vehicle's state and the feature map with associated features:

$$X_k = X_{k|k-1} + W * v \quad (17)$$

W is Kalman gain (formula (15)); v is the innovation calculated by formula (13). Covariance matrix P is:

$$P_k = P_{k|k-1} - WSW^T \quad (18)$$

2) Augment. This step is to add the new observed features into the map:

$$X_{aug} = \begin{bmatrix} X \\ z \end{bmatrix} \quad (19)$$

Convert feature z into global coordinate, that's:

$$g_i(x_v, z) = \begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} x_v + r \cos(\theta + \theta_v) \\ y_v + r \sin(\theta + \theta_v) \end{bmatrix} \quad (20)$$

Then linear transform X_{aug} , X and the covariance matrix P are:

$$X = f_i(X_{aug}) = \begin{bmatrix} X \\ g_i(x_v, z) \end{bmatrix} \quad (21)$$

$$P = \nabla f_{x_{aug}} P_{aug} \nabla f_{x_{aug}}^T \quad (22)$$

where $\nabla f_{x_{aug}}$ is Jacobian matrix with respect to the state X .

5. Experiment Results And Conclusion

5.1. Experiment results

The experiment was carried out in Tuandao bay of Qingdao and figure 4 shows the sonar image of it. Here, the blue color represents the features sonar observes, red color represents the vehicle's trajectory. In order to compare the sonar image with the real map, we combine the two by using Photoshop, as is shown in figure 5. From this figure, we can see that the sonar image is well matched with the satellite map from Google.

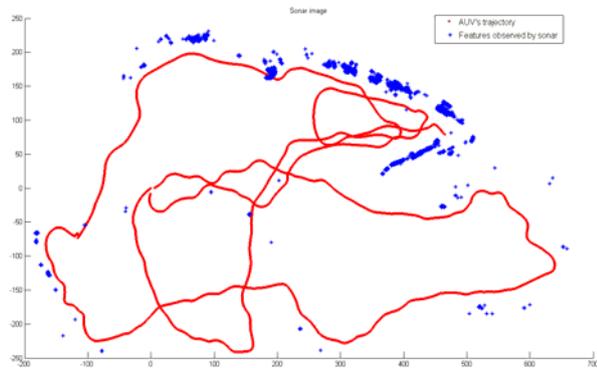


Fig. 4: Sonar image of Tuandao bay.



Fig. 5: Sonar image combined with satellite map in Tuandao bay.

Figure 6 shows comparison among the three trajectories: the red line represents SLAM's trajectory, the green one is the dead reckoning trajectory and the black one is the GPS trajectory. GPS trajectory is used as the standard here. As can be seen from the figure, compared with the green one, the red line is much closer to the black line.

Figure 7 is the trajectory error curve of SLAM and dead reckoning, both respect to the GPS trajectory. Clearly, the red curve (SLAM error curve) is always below the blue one (dead reckoning error curve). The maximum error of dead reckoning is about 95m while the maximum error of SLAM is only 57m. Figure 6 and figure 7 both indicate that the SLAM trajectory is more accurate than dead reckoning.

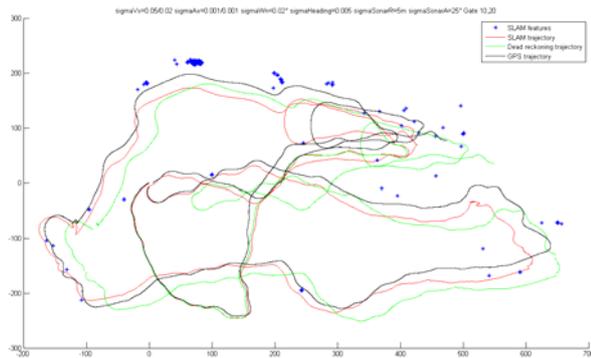


Fig. 6: SLAM, dead reckoning and GPS trajectory comparison.

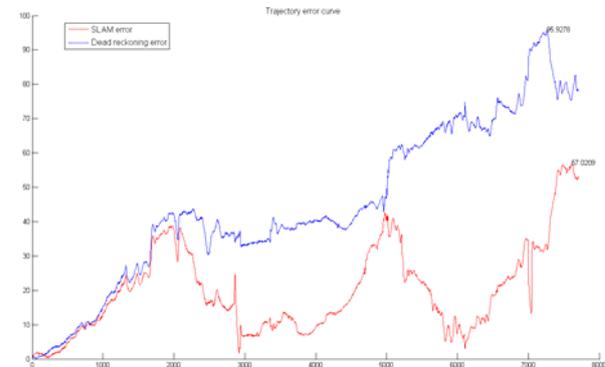


Fig. 7: Error curve of SLAM and dead reckoning, respect to GPS

5.2. Conclusion

The experiment results show that, with a forward-looking scanning sonar, SLAM can give a better trajectory and build a good environmental feature map. Precise location and environmental map will finally lead to underwater vehicle's autonomous navigation.

6. Acknowledgment

This research has been funded in part by National 863 Project (No.2006AA09Z231, No.2009AA12Z330), Science and Technology Public Relation Project of Shandong Province (No.2008GG1055011), and DR Fund of Shandong Province (BS2009HZ006).

7. References

- [1] David Ribas, "Report on the research project towards simultaneous localization & mapping for an AUV using an imaging sonar", pp. 6-13, Girona, June, 2005.
- [2] Trittech International Limited, "Notes for super seaking DST new generation digital CHIRP sonar", pp. 33-38, 2006.
- [3] Wenjing Wang, "Research on the applications of EKF-SLAM algorithm to the localization of underwater vehicles", pp. 28-35, Harbin Institute of Technology, 2007.
- [4] Jie Zhang, Bo He, Ke Yao, Yitong Wang, "Simultaneous localization and mapping for an autonomous underwater vehicle using forward-looking sonar", Control and Automation Publication Group, vol. 25, pp. 255-256, 2009.
- [5] Nini Yu, "Simultaneous localization and mapping for an AUV based on EKF", pp. 20-30, Ocean University of China, 2009.
- [6] Huixin Du, Weisheng Yan, Tao Zhao, "Simulation of forward looking sonar and obstacle avoidance for autonomous underwater vehicle", Torpedo Technology, vol. 18, Feb. 2010, pp. 49-50.
- [7] Yongqing Wang, "Research on data association for simultaneous localization and mapping", pp. 40-46, Ocean University of China, 2007.