



Article Research on an Underwater Target-Tracking Method Based on Zernike Moment Feature Matching

Wenhan Gao¹, Shanmin Zhou², Shuo Liu^{1,3,4,*}, Tao Wang^{1,4,*}, Bingbing Zhang¹, Tian Xia¹, Yong Cai² and Jianxing Leng⁵

- Key Laboratory of Ocean Observation-Imaging Testbed of Zhejiang Province, Zhejiang University, Zhoushan 316021, China; whgao@zju.edu.cn (W.G.); zhangbb@zju.edu.cn (B.Z.); xt0580@zju.edu.cn (T.X.)
- ² Ocean Research Center of Zhoushan, Zhejiang University, Zhoushan 316021, China; timoonboru@foxmail.com (S.Z.); caiyong888@zju.edu.cn (Y.C.)
- ³ Hainan Institute, Zhejiang University, Sanya 572025, China
- ⁴ The Engineering Research Center of Oceanic Sensing Technology and Equipment, Ministry of Education, Zhoushan 316000, China
- ⁵ Ocean College, Zhejiang University, Zhoushan 316000, China; jxleng@zju.edu.cn
- * Correspondence: shuoliu@zju.edu.cn (S.L.); twang001@zju.edu.cn (T.W.)

Abstract: Sonar images have the characteristics of lower resolution and blurrier edges compared to optical images, which make the feature-matching method in underwater target tracking less robust. To solve this problem, we propose a particle filter (PF)-based underwater target-tracking method utilizing Zernike moment feature matching. Zernike moments are used to construct the feature-description vector for feature matching and contribute to the update of particle weights. In addition, the particle state transition method is optimized by using a first-order autoregressive model. In this paper, we compare Hu moments and Zernike moments, and we also compare whether to optimize the particle state transition on the tracking results or not based on the effects of each option. The experimental results based on the AUV (autonomous underwater vehicle) prove that the robustness and accuracy of this innovative method is better than the other combined methods mentioned in this paper.

Keywords: underwater target tracking; Zernike moments; Hu moments; particle filter

1. Introduction

The PF is a commonly used state estimation method that has unique advantages for nonlinear processes and non-Gaussian models in practical problems [1,2]. It is currently widely used in scenarios such as target tracking, robot localization, etc. I. Masmitja [3] used a combination of PF and extended Kalman filter (EKF) to achieve the localization and tracking of an AUV, and they found that the PF provides better estimation for moving targets, while the EKF is more suitable for static targets.

Target characterization and recognition are the key aspects of tracking. In recent years, deep learning has been applied to target recognition [4], e.g., Shan Ma [5] proposed a method to solve the optimal feature combination based on GRNN (general regression neural network) judging criterion. The literature reference [6] used FCN (fully convolutional networks) to train and test an underwater sonar image data set and expressed the data set in a pixel matrix. However, due to the complexity of the underwater environment and noise overlay, dynamic target tracking still has urgent problems to be solved, such as harder target identification and poorer tracking accuracy. Manual feature-extraction methods are still of great importance.

Generally, we can characterize the target by the edges, contours, shapes, textures, regions, histograms, moment features, transformation coefficients, etc., in the images. However, in dynamic underwater observation, sonar images are of lower resolution and



Citation: Gao, W.; Zhou, S.; Liu, S.; Wang, T.; Zhang, B.; Xia, T.; Cai, Y.; Leng, J. Research on an Underwater Target-Tracking Method Based on Zernike Moment Feature Matching. *J. Mar. Sci. Eng.* **2023**, *11*, 1594. https:// doi.org/10.3390/jmse11081594

Academic Editor: Sergei Chernyi

Received: 14 July 2023 Revised: 10 August 2023 Accepted: 11 August 2023 Published: 14 August 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). are more susceptible to noise interference compared to optical images. The relative motion between the sonar and the target can easily lead to distortions in target imaging [7,8]. Therefore, a single-feature description method such as contour or texture is less accurate. There are many scholars who use histogram methods, such as Songwei Huang [9], who used a color-feature histogram to construct feature-description vectors for sonar images, and Xiao Wang [10], who extracted a color histogram and the shape texture to describe features together. The histogram can reflect the probability distribution of pixel values but lacks the target spatial location and shape information. In recent years, the multifeature fusion method has often been used to solve the tracking of moving targets in cluttered environments. The color features, edge features, and direction features of the target are the more commonly fused features. This method helps to improve the tracking accuracy, but fusing more features will enhance the complexity of the algorithm and affect the tracking in real time [11,12].

In this paper, we study the application of invariant moments in underwater target tracking. Hu moments, Zernike moments, Radon moments, etc., all belong to statistical feature-description methods. Because of the invariance in terms of translation, rotation, and scale change, these methods are suitable for image matching, pattern recognition, and other fields. Tiedong Zhang [13] designed an observation model in a PF tracker by fusing the area features and Hu moment features of the target region. Ziqi Wang [14] and Ji Li [15] conducted experiments in an indoor pool, and they demonstrated that using the Hu moment method can obtain better tracking results than the grayscale template matching method. However, these experiments had fewer disturbances and lacked the dynamic process of more outdoor environments. Hengguang Li [16] compared the performance of different moments on processing the sonar images of marine organisms and demonstrated that Zernike moments are more advantageous than Hu and Radon moments for static target recognition, but the robustness of dynamic target recognition was not further explored.

Hu invariant moments consist only of a nonlinear combination of second- or thirdorder normalized central moments. In contrast, Zernike moments are defined in the unit circle, and they are orthogonal moments and outperform Hu moments in terms of information redundancy. Zernike low-order moments can describe the overall shape of the target, and the extracted features are less correlated and more noise-resistant. In addition, arbitrary higher-order moments can be constructed to describe the image details [17]. In this paper, in order to study the robustness and accuracy of invariant moment features in dynamic target recognition, we use Zernike moments to describe the features and construct the observation model based on the assumption of an approximate 2D tracking scenario. Also, the Hu moment method is implemented and compared.

This paper contains four sections. The first section discusses the origin of the targettracking method. The second section introduces the hardware platform and algorithm design of the tracking system in the PF framework. The third section presents the experimental and data analysis results, and demonstrates the feasibility and robustness of the algorithm. The fourth section provides a conclusion to the whole paper and an outlook for the future.

2. AUV Target-Tracking System Based on Mechanical Scanning Sonar

2.1. Hardware Architecture

The target-tracking sensor system of the AUV in this paper mainly includes an IM-AGENEX MODEL 881L mechanical scanning sonar, HsKINS-SG-4500C1 fiber-optic gyro, PATHFINDER DVL, altimeter, etc. The sensor distribution of the AUV tracking system is shown in Figure 1.



Figure 1. Sensor distribution of the AUV tracking system.

The observation module consists of a mechanical scanning sonar. The navigation module consists mainly of a fiber-optic gyro, DVL, and altimeter, and this module calculates the position and attitude information of the AUV. The control module is position closed-loop and velocity closed-loop. We fuse the position and attitude information of the AUV to correct the distortion of a sonar image, then we obtain the target information in an odometry (referred to as odom in the latter text) coordinate system, which is then inputted into the PF tracker (Figure 2).



Figure 2. The relationship between the several modules of the tracking system.

2.2. Mechanical Scanning Sonar Data Preprocessing

2.2.1. Data Preprocessing

The mechanical scanning sonar is a single-beam sonar which has the characteristics of simpler structure, lower cost, and lower power consumption compared to the multibeam sonar, and it can easily be mounted on a small AUV. The echo data of each beam comprise a Ping, and each Ping contains 500 points to record the echo information. Each point is recorded as a Bin, which contains the position coordinates and echo intensity information. In the sonar coordinate system, when we analyze the position and intensity information of the 500 points of each echo beam, we use PCL (Point Cloud Library) to construct a point cloud to express the echo information. In this paper, the navigation module error is 0.7%

of the distance. Since the research in this paper focuses on the application of invariant moments to dynamic target feature recognition, we designed the time and distance scales to be small in the experiment so that the positioning drift error can be ignored.

The target detection is performed by the process in Figure 3. After obtaining the point cloud data in the odom coordinate system, we can extract the position of the target to be tracked. The position information of the target is described by the centroid coordinates of the extracted target point cloud.



Figure 3. Target detection process.

2.2.2. Calibration

In order to obtain the observation data in the odom coordinate system, a coordinate transformation from sonar coordinate system to base_link carrier coordinate system and then to odom coordinate system is required. In this paper, the base_link coordinate system and the fiber-optic gyro coordinate system are considered to be the same, and the base_link *X* axis passes through the center of the sonar installation position. The dynamic transformation relationship between the base_link and odom is obtained from the navigation module. The transformation relationship between the sonar coordinate system and the base_link is static, and its translational transformation can be approximated by the sensor mounting position; as for the rotation relationship, the calibration of the horizontal deflection β of the two coordinate systems is required (Figure 4).



Figure 4. The model of sensor misalignment calibration (In the NED coordinate systems).

The AUV needs to observe obstacle *A* at two positions *B* and *C*, and the yaw angles at two positions are similar (the error within 5° can be ignored). There are three sets of data needed to solve β :

• The absolute coordinates (*x*_{*B*}, *y*_{*B*}) and (*x*_{*C*}, *y*_{*C*}) of the AUV at *B* and *C*, obtained from GPS;

- The relative coordinates (*x*_A, *y*_A) of target *A* at *B* and *C* positions under the sonar coordinate system;
- The yaw angles at *B* and *C*. Equations (1)–(5) are based on the triangular relationship in Figure 4:

$$\gamma = \arctan\left(\frac{y_C - y_B}{x_C - x_B}\right),\tag{1}$$

$$\angle C'B'A = \arccos\left(\frac{a^2 + c^2 - b^2}{2ac}\right),\tag{2}$$

$$\alpha = \arctan\left(\frac{y_A}{x_A}\right),\tag{3}$$

$$\theta = \alpha - \angle C'B'A,\tag{4}$$

$$\beta = \gamma - yaw - \theta. \tag{5}$$

The value of β is -2.70° after performing statistical processing for multiple sets of experiments. The transformation matrix for converting points under the sonar coordinate system to the base_link is

$$M = \begin{bmatrix} \cos(\beta) & -\sin(\beta) & 0 & 0.505\\ \sin(\beta) & \cos(\beta) & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.9989 & 0.0471 & 0 & 0.505\\ -0.0471 & 0.9989 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1 \end{bmatrix}.$$
 (6)

Finally, the dynamic conversion from the base_link to the odom coordinate system is realized by fusing the position and attitude information of the AUV in real time.

2.3. Algorithm Design

2.3.1. Feature-Description Vector Construction

Target detection requires the identification and localization of a specific target object from an image, and the previous target detection session relies on prior information to obtain the position of the target of interest. Feature extraction is a multidimensional mathematical description of the target, which in turn realizes the automatic identification of the target and is the basis of target detection. In this paper, we have implemented both Zernike moment- and Hu moment- [18] based feature matching methods and also compared the robustness and accuracy of both.

The regular moments project the image function f(x, y) onto the base of $x^p y^q$. Further, we can extend the base to the more general polynomial base $P_p(x)P_q(y)$. Zernike proposed a set of orthogonal polynomials $\{V_{nm}(x, y)\}$, which are orthogonal in the unit circle $\{x^2 + y^2 \le 1\}$. Zernike's nth-order polynomial $\{V_{nm}(x, y)\}$ is defined as

$$V_{nm}(x,y) = V_{nm}(\rho,\theta) = R_{nm}(\rho)\exp(jm\theta), \tag{7}$$

where $n = 0, 1, 2, ..., \infty$, and *m* takes on positive and negative integer values subject to the following conditions:

$$n - |m| \text{ is even,} \quad |m| < n, \tag{8}$$

In Equation (7), ρ is the length of the vector from the origin to the pixel point (x, y), and -1 < x, y < 1. θ is the angle between the vector ρ and the *X* axis. $R_{nm}(\rho)$ is a radial polynomial, which can be expressed as

$$R_{nm}(\rho) = \sum_{s=0}^{n-|m|/2} (-1)^s \frac{(n-s)!}{s! \left(\frac{n+|m|}{2}-s\right)! \left(\frac{n-|m|}{2}-s\right)!} \rho^{n-2s} .$$
(9)

M.R. Teague proposed Zernike moments based on the theory of orthogonal polynomials [19]. For a two-dimensional image function f(x, y), the nth-order Zernike moment with repetition rate *m* is defined as

$$Z_{nm} = \frac{n+1}{\pi} \iint_{x^2 + y^2 \le 1} V_{nm}^*(x, y) f(x, y) dx dy, \quad x^2 + y^2 \le 1.$$
(10)

where * indicates that the conjugate is taken, and the above equation is expressed in the polar coordinate system as

$$Z_{nm} = \frac{n+1}{\pi} \int_0^1 \int_0^{2\pi} R_{nm}(\rho) e^{-jm\theta} f(\rho,\theta) \rho d\rho d\theta.$$
(11)

For a two-dimensional image whose Zernike moments are complex numbers and whose real and imaginary parts are denoted as C_{nm} and S_{nm} , respectively:

$$C_{nm} = \frac{2n+2}{\pi} \int_0^1 \int_0^{2\pi} R_{nm}(\rho) \cos(m\theta) f(\rho,\theta) \rho d\rho d\theta,$$
(12)

$$S_{nm} = -\frac{2n+2}{\pi} \int_0^1 \int_0^{2\pi} R_{nm}(\rho) \sin(m\theta) f(\rho,\theta) \rho d\rho d\theta.$$
(13)

The image analysis in practical problems requires a discretization of Equation (11):

$$Z_{nm} = \frac{n+1}{\pi} \sum_{x} \sum_{y} f(x,y) V_{nm}^*(\rho,\theta), \quad x^2 + y^2 \le 1.$$
(14)

In the calculation of Zernike moments, the center of the pixel point in the target image region must be used as the origin (position normalization), and the pixel coordinates are projected into the unit circle (scale normalization). In this paper, a transformation method for solving Zernike moments proposed in the literature [20] is learned to solve the moments for the sonar point cloud.

The feature vector T constructed with the seven Hu moment values is seven-dimensional, i.e., $T = [I_1, I_2, ..., I_7]$, while the dimensionality of the feature vector constructed with Zernike moments is related to the values of m and n. Therefore, we take the combination of m and n values as in Table 1 to construct the seven-dimensional vector of Zernike invariant moments in order to let it be in contrast with the Hu moments.

Table 1. Combination of <i>m</i> and <i>n</i> corresponding to Zernike moments.	

	I_1	I_2	I_3	I_4	I_5	I_6	I_7
т	1	3	3	5	3	5	7
п	3	5	7	7	9	9	9

2.3.2. Tracking System Initialization

The basic idea of the PF is to represent the posterior probability distribution with a series of random state samples obtained from the posterior. The PF algorithm estimates the statistical properties of random variables based on the state and respective weights of the samples after the samples pass through the nonlinear system. The samples here

are the so-called particles, and a particle is a possible hypothesis based on the real-world state at moment *t*. The PF algorithm consists of initialization, state transition, update, and resampling steps [21], and its algorithmic flow is shown in Figure 5.



Figure 5. Framework of PF algorithm.

In this paper, we approximate the tracking scenario as two-dimensional, considering only the position information as a state quantity. The position is written as $P = [P_x, P_y]^T$, where P_x and P_y are the coordinates of the target centroid position in the odom coordinate system.

Firstly, we construct the feature template using the first three frames of the sonar point cloud. Considering that the Zernike moments are solved on the orthogonal basis of the unit circle and that the tracking target is blocky and not elongated, a circle is used to construct the feature template. We calculate the farthest distance of the target point cloud to be tracked from the segmentation and approximate the distance value as the template's radius *r* (after calculation, this paper takes *r* = 1 m). Then, we can calculate three feature-description vectors of the target in each of the three frames and construct the template feature vector using the mean values of the corresponding elements of the three vectors. Further, we can initialize the target position $P^{init} = \left(P_x^{init}, P_y^{init}\right)$ using the motion state of the target in the third frame and then randomly initialize particles within a certain range of the target. Finally, the individual particle positions are obtained as

$$P^{i} = \left(P_{x}^{i}, P_{y}^{i}\right), \quad i = 1, 2, \dots N.$$

$$(15)$$

where *N* is the total number of particles, which is set to 120 in this paper. P_x^i and P_y^i are expressed as

$$P_x^i = P_x^{init} + \delta, \tag{16}$$

$$P_y^i = P_y^{init} + \delta, \tag{17}$$

where δ is a random number obeying the Gaussian distribution of $(0, r^2)$, and the initial particle weights are

$$w_0^{(1)} = w_0^{(2)} = \dots = w_0^{(N)} = \frac{1}{N}$$
 (18)

When subsequent point clouds arrive, the positions and weights of the particles are updated based on the state transition model and the observation model.

2.3.3. State Transition Model

PF tracking algorithms are mostly based on the smooth motion assumption in the state transition process, i.e., no abrupt changes occur during the target motion. The literature [22] reviewed the application of PF algorithms in smooth motion tracking. For the case of drastic changes in target appearance and attitude, a PF tracking algorithm based on a memory mechanism was proposed in the literature [23], which used historical target state sequences to achieve the simultaneous estimation of target position and attitude, but this algorithm has a high spatial complexity. In the real environment, the motion of the target has more autonomy, which is difficult to accurately express using mathematical modeling. The first-order autoregressive model [13–15] is often used for state transition analysis. This paper draws on this idea and also optimizes the state transition process for the possible degradation of the particle weights.

Only considering the position state of the target, the first-order autoregressive model is used to model the following:

$$P_k = AP_{k-1} + Bw_{k-1}.$$
 (19)

For each particle, we have

$$P_{x(k)}^{i} = A_{x}P_{x(k-1)}^{i} + B_{x}w_{k-1}, \quad i = 1, 2, \dots N.$$
⁽²⁰⁾

$$P_{y(k)}^{i} = A_{y}P_{y(k-1)}^{i} + B_{y}w_{k-1}, \quad i = 1, 2, \dots N.$$
⁽²¹⁾

To simplify the state transition model, $A_x = A_y = 1$, $B_x = B_y = 1$ are taken in Equations (20) and (21), i.e., the current position is the previous moment's position plus the Gaussian term w_{k-1} . w_{k-1} is a random number obeying Gaussian distribution of $(0, r^2)$. Gaussian variance is selected by considering the maximum edge distance of the tracked target and its motion speed (about 0.1 m/s in this experiment).

The above-mentioned particle state transition method is the Gaussian random method for each frame of observation. For a dynamic target with autonomous motion, this method generates a large number of particles that move in the opposite direction of the target motion. Therefore, this method could reduce the ability of the particles to "catch" the target, cause degradation of the particle weights, and reduce the efficiency of the algorithm. Thus, we propose an optimized particle state transition method in contrast to the above method.

At the state transition of each frame, the first two position estimations are considered together. The angle between the line connecting two adjacent position points and the *X* axis is α , as in Figure 6.



Figure 6. Angle α at two adjacent positions.

We define two modes of particle state transition and note that the distance between the positions of two adjacent frames is *L*. Mode 1: If $L \le 0.3$ m, we consider that the target motion is slow and use the ordinary particle state transition method outlined in the previous section. Mode 2: If L > 0.3 m, we consider that the target keeps a faster motion and let the particles tend to the direction of the target motion with a certain probability (which is 0.5) when performing particle state transition. Immediately after, we generate Gaussian random numbers G_x and G_y , obeying $(0, r^2)$ distribution, in both directions X and Y and compute $|\alpha|$:

$$\Delta_x = Target_k(x) - Target_{k-1}(x), \tag{22}$$

$$\Delta_y = Target_k(y) - Target_{k-1}(y), \tag{23}$$

$$|\alpha| = |\arctan(\frac{\Delta_y}{\Delta_x})|.$$
(24)

If $|\alpha| < 30^{\circ}$ or $|\alpha| > 150^{\circ}$, the target is considered to be moving more towards the *X* axis. We can adjust G_x and G_y so that the larger absolute value corresponds to the motion in the X direction and set G_x and G_y to have the same sign as Δ_x and Δ_y , respectively. If $|\alpha| > 60^{\circ}$ and $|\alpha| < 120^{\circ}$, the target is considered to be moving more towards the *Y* axis. We can adjust G_x and G_y so that the larger absolute value corresponds to the motion in the Y direction and set G_x and G_y to have the same sign as Δ_x and Δ_y , respectively.

2.3.4. Observation Model

For each frame, we take *N* particle samples $\{x_{t-1}^{(i)}\}$ from the prior distribution $p(x_t | x_{t-1}^{(i)})$, then obtain a point cloud of the same range as the initialized template at each particle position after state transition and calculate the feature vector of this local point cloud. Finally, we can update the weight of each particle according to its similarity with the template feature vector.

For the feature vectors constructed with Zernike or Hu moments, the data are processed in the same way. To facilitate the analysis, each moment value is first transformed as

$$I' = abs(\lg(|I|)). \tag{25}$$

Then, the cosine formula is used to obtain the similarity of the current feature vector *A* and the template feature vector *B*. Define the cosine value as *Sim*:

$$Sim = \frac{A \cdot B}{|A| \cdot |B|},\tag{26}$$

The closer the *Sim* is to 0, the more similar the two vectors are. In order to compare similarity in linear dimension, every obtained *Sim* value is inverted to a corresponding vector angle α :

$$\alpha = \arccos(Sim), \ \alpha > 0. \tag{27}$$

The smaller the α is, the more similar the two point clouds are, and the corresponding particle weight should be larger. In order to avoid the impact of extreme differences on the algorithm performance, a threshold $\alpha_{threshold}$ is set. In the first calculation of particle weights, the α values corresponding to all particles are arranged in an array from smallest to largest, and the value at 2/3 of the total number of particles from the minimum α is taken as the threshold $\alpha_{threshold}$ of the whole tracking process. Then, we use the minimum absolute difference (MAD) function to calculate the similarity value $MAD^{(i)}$ for each particle. A

particle with α greater than $\alpha_{threshold}$ is considered to be too different from the template, and its weight is set to 0:

$$MAD^{i} = \begin{cases} \frac{\alpha_{threshold} - \alpha^{(i)}}{\alpha_{threshold}}, & \alpha^{(i)} < \alpha_{threshold} \\ 0, & \alpha^{(i)} \ge \alpha_{threshold} \end{cases}$$
(28)

The probability distribution based on the angular similarity can be defined as

$$p\left(z_t \middle| x_t^{(i)}\right) \propto \exp\left(-\frac{MAD^{(i)}}{2\sigma^2}\right),$$
 (29)

where σ is a constant. Thus, the particle weights based on sequence importance sampling are updated as

$$w_t^{(i)} = p\left(z_t \middle| x_t^{(i)}\right) w_{t-1}^{(i)}.$$
(30)

After normalizing the particle weights, we sum the state weights to obtain an estimate of the target position in the current frame (Equation (31)) and wait for the next frame to arrive.

$$E_{x_t \sim p(x_t|z_{1:t})}[x_t] = \frac{1}{N} \sum_{i=1}^N x_t^{(i)} \widetilde{w}_t^{(i)}.$$
(31)

2.3.5. Resampling

The resampling method is designed to solve the particle weight degradation problem that occurs during the iterative process of the PF. This method can effectively allow the PF to avoid wasting arithmetic power on particles with tiny weights. The essence is to copy the particles with large weights, eliminate the particles with small weights, and assign the same weights to the new particles. Define the effective sampling scale:

$$Neff \approx \frac{1}{\sum_{i=1}^{N} (w_t^{(i)})^2}$$
 (32)

After updating the weights of particles using each frame's data, if the *Neff* value is less than 2/3 of the total number of initial particles, it is considered necessary to perform particle resampling. When resampling, Gaussian random numbers are added to the positions of the new particles obtained by replication, with the aim of enhancing particle diversity. After resampling, the weights of all new particles are 1/N.

2.3.6. Algorithm

Algorithm 1 in the following table is the PF underwater target-tracking process based on mechanical scanning sonar.

The first row is for template initialization and particle set initialization. Rows 2 to 14 iteratively solve for the target position when the sonar sequence observation point clouds arrive. Among them, rows 3–7 update the state transition and weights of the particles, rows 8–11 calculate the normalized weights of the particles, and row 12 calculates the target position estimate at the current moment, while row 13 resamples the particles.

3. Experiments and Results

3.1. Experimental Scenes

We conducted AUV target-tracking experiments in late spring in Maishan Reservoir, Zhoushan, which is a freshwater reservoir. The reservoir is surrounded by a dike on the west side and mountains on the remaining three sides. The water is wide, and it had a temperature of about 8 °C, calm wind, and calm waves at the time of the experiments, which provided the experimental conditions. For the experiments, a white plastic bucket was chosen to simulate a dynamic obstacle with a diameter of 780 mm and a height of 1000 mm, and a 14 kg iron block was tied to the bucket so that it could be suspended in the water. The iron block was approximately on the axis of the bucket in about 2.5 m of water. The experimental layout is shown in Figure 7.

Algorithm 1: Particle Filter Underwater Target Tracking

INPUT: Sonar sequence point clouds **OUTPUT:** Position estimation of AUV, $E_{x_t \sim p(x_t|z_{1:t})}[x_t]$ 1: Initialize feature template, $\chi_t = \emptyset$ 2: While (Sonar sequence point clouds arrive) do **FOR** i = 1 : N3: sample $x_t^{(i)} \sim p(x_t | x_{t-1}^{(i)})$ 4: $w_t^{(i)} = p\left(z_t \middle| x_t^{(i)}\right) w_{t-1}^{(i)}$ $\chi_t = \chi_t + \left\langle x_t^{(i)}, w_t^{(i)} \right\rangle$ 5: 6: 7: **END FOR** $t = \sum_{i=1}^{N} w_t^{(i)}$ 8: **FOR** i = 1 : N9: $\widetilde{w}_t^{(i)} = w_t^{(i)}/t$ 10: END FOR 11:
$$\begin{split} & E_{x_t \sim p(x_t|z_{1:t})}[x_t] = \frac{1}{N} \sum_{i=1}^{N} x_t^{(i)} \widetilde{w}_t^{(i)} \\ & \left[\left\{ x_t^{(i)}, w_t^{(i)} \right\}_{i=1}^{N} \right] = Resample \left[\left\{ x_t^{(i)}, w_t^{(i)} \right\}_{i=1}^{N} \right] \end{split}$$
12: 13: 14: END WHILE



Figure 7. Experimental scene diagram. (**a**) Design path on the satellite map (as indicated by the white arrow). (**b**) AUV placement. (**c**) Target bucket placement.

A rubber boat equipped with a propeller hookup was used to drag the bucket to the shore in slow motion (due to water resistance and the impact of the flow, the actual path was not straight). In order to make an accuracy assessment of the algorithm results, a differential GPS module was placed directly above the bucket to read the position change, which was used as the a priori position to compare with the estimated position. The AUV was continuously observed behind the target, with a depth of about 1.5 m throughout (Figure 8).

Based on the above scenarios, we conducted three different sets of experiments to verify the robustness and accuracy of the algorithms in this paper:

- 1. Static target with AUV hovering observation;
- 2. Dynamic target with AUV hovering observation;
- 3. Dynamic target with AUV following observation.

Combining the characteristics of this mechanical scanning sonar and the application scenario of this paper, we set the parameters of the sonar to what is shown in Table 2.



Figure 8. Tracking process, AUV behind the target.

Table 2. The main parameters of the sonar.

Parameter	Value
Frequency	675 kHz
Pulse length	110 μs
Start gain	10 dB
Range	40 m
Step size	1.2°
Sector width	120°

In this paper, a laptop equipped with an Intel Core i5 2.40 GHz processor was used to conduct the data offline processing. We built the PF tracking algorithm system in C++ based on the ROS (Robot Operating System) platform. The point cloud data obtained from sonar detection was interconnected with other modules of the AUV through the "Topic" communication mechanism in ROS.

3.2. Data Analysis

3.2.1. Static Target with AUV Hovering Observation

In this set of static target data, the target to be tracked in the first three frames of the sonar scan point cloud is extracted (Figure 9), and it is used to construct the feature template. The white dashed circle indicates the size of the template, which surrounds the target to be tracked.



Figure 9. Target in the first three frames of the first set of tracking data. (a) Sonar scan point cloud. (b) The first frame. (c) The second frame. (d) The third frame.

We take the Hu moments of the above three frames and construct a 7-dimensional feature-description vector $T = [I_1, I_2, ..., I_7]$ using the mean values of the corresponding elements. The individual elements of the feature-description vector are shown in Table 3. Then, the same operation is performed for the Zernike moments.

Template Features Moments	Hu	Zernike
I_1	1.8182	1.3843
I_2	5.4463	3.9044
I_3	6.4658	7.1292
I_4	8.7426	10.1156
I_5	16.5203	12.8892
I_6	12.0946	16.6743
I ₇	16.6599	20.3906

Table 3. Different moment feature-description vectors for the first set of tracking data.

Figure 10 shows the tracking process after using the Zernike moment feature-description method and optimized state transition. The white point set is the set of position estimation points, and the green point set is the particle population distributed at the current moment. Since the target to be tracked is static, the position estimation changes very little.



Figure 10. The tracking process for the first set of tracking data. Only the method of taking Zernike moments and optimized state transition is shown. (a) First three frames. (b) First ten frames. (c) Till the end.

Due to the small distance between the position estimation points of two adjacent moments of this static target (L < 0.3 m in most cases), the algorithm will enter mode 1 under the optimized state transition method. The tracking deviations (the difference between the GPS values and the estimated values) of the PF tracker in both the X and Y directions are shown in Figure 11.



Figure 11. Tracking deviations of the first set of tracking data. (**a**) X direction deviations. (**b**) Y direction deviations.

Define the mean value of the distance deviation *S* as

$$S = \frac{1}{N} \sum_{i=1}^{N} \sqrt{\left(X_{error}^{i}\right)^{2} + \left(Y_{error}^{i}\right)^{2}}$$
(33)

Further, we calculate the means and variances of the deviations in both the X and Y directions for the first data set, as in Table 4.

Table 4. Means and variances of the first data s

Deviation	Hu	Zernike
X deviation mean/m	0.0492	-0.0291
X deviation variance/m ²	0.0754	0.0187
Y deviation mean/m	-0.1559	0.0292
Y deviation variance/m ²	0.0711	0.0238
S/m	0.3791	0.1883

For the static target in the first set of tracking data, it can be seen from the above analysis that the feature-description method using Zernike moments is better. The absolute values of means and variances in the X and Y directions are smaller in this method, where the mean value is about 41% and 81% smaller than the Hu moment method, respectively, and the S value is about 50% smaller.

3.2.2. Dynamic Target with AUV Hovering Observation

Similar to the previous section, for this dynamic target data set, we take the first 3 frames of the target to construct the feature template (Figure 12).



Figure 12. Target in the first three frames of the second set of tracking data. (**a**) Sonar scan point cloud. (**b**) The first frame. (**c**) The second frame. (**d**) The third frame.

The corresponding feature-description vectors are obtained as in Table 5.

Table 5. Different moment feature-description vectors for the second set of tracking data.

Template Features Moments	Hu	Zernike
	2.5634	0.1720
I_2	7.7801	3.0683
I_3	9.5124	5.9074
I_4	10.2396	9.5158
I_5	21.8039	12.3790
I_6	14.9056	15.5995
I ₇	20.2795	19.2436

Figure 13 shows the tracking process after using the Zernike moment feature-description method and optimized state transition method. The white point set is the set of position estimation points, and the green point set is the particle population distributed at the current moment. In the optimized state transition method, the algorithm almost always goes to mode 2 and obtains a set of position estimation points. Figure 14 shows the differential GPS path of the target and the tracking paths under several methods.



Figure 13. The tracking process for the second set of tracking data. Only the method of taking Zernike moments and optimized state transition is shown. (a) First three frames. (b) First ten frames. (c) Till the end.





The tracking deviations of the PF tracker in both the X and Y directions are shown in Figure 15.



Figure 15. Tracking deviations of the second set of tracking data. (**a**) X direction deviations. (**b**) Y direction deviations.

Further, we calculate the means and variances of the deviations in both the X and Y directions for the second data set, as in Table 6.

Deviation	Hu Non-Optimized	Hu Optimized	Zernike Non-Optimized	Zernike Optimized
X deviation mean/m	9.0516	1.0108	1.8659	0.0221
X deviation variance/m ²	39.5647	0.4556	1.9981	0.0798
Y deviation mean/m	3.0310	0.6296	0.6589	0.2486
Y deviation variance/m ²	6.3450	0.1157	0.7506	0.0807
S/m	9.6144	1.2914	2.0941	0.4059

Table 6. Means and variances of the second data set.

For the second data set, it can be seen from Figure 15 and Table 6 that better tracking accuracy is obtained by using Zernike moments under the optimized state transition condition. The absolute values of the means of deviations in the X and Y directions in the Zernike moment method are about 98% and 61% smaller than those in the Hu moment method, respectively. The S value is nearly 69% smaller, and the corresponding variances are smaller, too. In addition, the Hu moment method quickly fails to track when the state transition is not optimized, while the Zernike moment method still maintains stable tracking. On the other hand, from the moments perspective, in the Hu moment method, stable tracking is achieved only after the optimized state transition. However, in the Zernike moment method, stable tracking is achieved whether the state transition is optimized or not. The absolute values of the means of the deviations in the X and Y directions after optimization are about 98% and 62% smaller than those without optimization, respectively, in the Zernike moment method, and the distance mean S is about 81% smaller; the corresponding variances are smaller, too.

3.2.3. Dynamic Target with AUV following Observation

In the same way, we extract the first three frames to construct the feature template, as shown in Figure 16.



Figure 16. Target in the first three frames of the third set of tracking data. (**a**) Sonar scan point cloud. (**b**) The first frame. (**c**) The second frame. (**d**) The third frame.

The corresponding feature-description vectors are obtained in Table 7.

Figure 17 shows the tracking process after using the Zernike moment feature-description method and optimized state transition. The white points are the position estimation points, the green point set is the particle population distributed at the current moment, and the purple line is the motion path of the AUV. For the optimized state transition method, the algorithm almost always goes to mode 2 and obtains a set of position estimation points. Figure 18 shows the differential GPS path of the target and the tracking paths under several methods.

Template Features Moments	Hu	Zernike
I_1	2.5622	0.1983
I_2	7.0985	2.7418
I_3	8.6391	5.9358
I_4	9.3487	9.1654
I_5	18.3708	12.9062
I_6	13.3045	15.3804
I_7	19.2669	18.5920

Table 7. Different moment feature-description vectors for the third set of tracking data.



Figure 17. The tracking process for the third set of data. Only the method of taking Zernike moments and optimized state transition is shown. (a) First three frames. (b) First twenty frames. (c) Till the end.



Figure 18. The differential GPS path and tracking paths of the target in the third data set.

The tracking deviations of the PF tracker in both the X and Y directions are shown in Figure 19.

Further, we calculate the means and variances of the deviations in both the X and Y directions for the third data set, as in Table 8.



Figure 19. Tracking deviations of the third set of tracking data. (**a**) X direction deviations. (**b**) Y direction deviations.

Deviation	Hu Non-Optimized	Hu Optimized	Zernike Non-Optimized	Zernike Optimized
X deviation mean/m	6.3420	0.6765	0.7829	0.5286
X deviation variance/m ²	22.7151	0.4830	0.2180	0.3365
Y deviation mean/m	-1.7531	-0.4954	-0.3845	-0.3170
Y deviation variance/m ²	1.7965	0.3502	0.2768	0.2386
S/m	6.7260	1.1086	1.0502	0.8761

Table 8. Means and variances of the third data set.

Similar to the second data set, for the third data set, it can be seen from Figure 19 and Table 8 that better tracking accuracy is obtained by using Zernike moments under the optimized state transition condition. The absolute values of the means of deviations in the X and Y directions in the Zernike moment method are about 22% and 36% smaller than those in the Hu moment method, respectively. The S value is nearly 21% smaller, and the corresponding variances are smaller. In addition, the Hu moment method quickly fails to track when the state transition is not optimized, while the Zernike moment method still maintains stable tracking. On the other hand, from the moments perspective, in the Hu moment method, stable tracking is achieved only after the optimized state transition. However, in the Zernike moment method, stable tracking is achieved whether the state transition is optimized or not. The absolute values of the means of the deviations in the X and Y directions after optimization are about 32% and 18% smaller than those without optimization, respectively, in the Zernike moment method; the distance mean S is about 17% smaller and the corresponding variances are smaller, too.

4. Conclusions

In this paper, we study the PF target-tracking method based on mechanical scanning sonar. Through AUV tracking experiments and data analysis, we demonstrate that the method of Zernike moment feature matching leads to better tracking results. In addition, the optimization of the first-order autoregressive model using target motion convergence is also beneficial to the accuracy of the algorithm. This paper further corroborates the view in the literature [16] that Zernike moments are superior to Hu moments in target identification.

As we can see, the experimental data are processed offline for comparison, and we are ready to carry out online real-time tracking experiments in the future. At the same time, we will optimize the case of unstable observation caused by disruption to improve the tracking accuracy of a single target.

Author Contributions: Conceptualization, S.L., T.W. and J.L.; methodology, W.G., S.Z., S.L. and T.W.; software, W.G., S.Z. and B.Z.; investigation, W.G., S.Z., S.L., T.W., B.Z., T.X., Y.C. and J.L.; validation, W.G., S.L. and T.W.; data curation, S.Z. and W.G.; formal analysis, W.G., S.Z., S.L. and T.W.; writing—original draft preparation, W.G., S.Z., B.Z. and T.X.; writing—review and editing, W.G., S.Z., S.L., T.W., B.Z., T.X., Y.C. and J.L.; visualization, T.X.; supervision, Y.C.; project administration, Y.C.; funding acquisition, S.L.; resources, S.L. and J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the "Pioneer" and "Leading Goose" R&D Program of Zhejiang (2022C03041 and 2023C03124), the Research Program of Sanya Yazhou Bay Science and Technology City (SKYC2020-01-001), the Strategic Priority Research Program of the Chinese Academy of Sciences (XDA22040202), and "the Fundamental Research Funds for the Central Universities" + "226-2023-00049".

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

- 1. Barfoot, T.D. State Estimation for Robotics; Gao, X., Xie, X.J., Eds.; Xi'an Jiaotong University Press: Xi'an, China, 2018.
- Handschin, J.E. Monte Carlo Techniques for Filtering and Prediction of Nonlinear Stochastic Processes; University of London: London, UK, 1968.
- Masmitja, I.; Bouvet, P.J.; Gomariz, S. Underwater mobile target tracking with particle filter using an autonomous vehicle. In Proceedings of the OCEANS, Aberdeen, UK, 19–22 June 2017; pp. 1–5.
- Tan, P.L.; Wu, X.B.; Zhang, X.Y. Review on Underwater Target Recognition Based on Sonar Image. *Digit. Ocean. Underw. Warf.* 2022, 5, 342–353.
- 5. Ma, S. Multi-Target Tracking of AUV Based on Forward Looking Sonar; Harbin Engineering University: Harbin, China, 2016.
- Nguyen, H.T.; Lee, E.; Bae, C.H. Multiple object detection based on clustering and deep learning methods. Sensors 2020, 20, 4424. [CrossRef] [PubMed]
- 7. Li, Q.W.; Huo, G.Y.; Zhou, Y. Sonar Image Processing; Science Press: Beijing, China, 2015.
- 8. Gong, W.J.; Tian, J. Underwater sonar image small target recognition method based on shape features. *J. Appl. Acoust.* **2021**, *40*, 294–302.
- 9. Huang, S.W.; Zhu, Z.T. A sonar image target tracking algorithm based on particle filter. *Ship Sci. Technol.* 2019, 41, 135–139.
- 10. Wang, X.; Zou, Z.W. Target Detection in Colorful Imaging Sonar Based on Multi-feature Fusion. Comput. Sci. 2019, 46, 177–181.
- 11. Zan, M.E.; Zhou, H. Survey of Particle Filter Target Tracking Algorithms. *Comput. Eng. Appl.* 2019, 55, 8–17.
- 12. Liu, T.J. Adaptive hierarchical particle filter in dynamic tracking scenarios. J. Electron. Compon. Inf. Technol. 2018, 1, 17–21.
- 13. Zhang, T.D.; Wan, L. Underwater Object Tracking Based on Improved Particle Filter. J. Shanghai Jiaotong Univ. 2012, 46, 943–948.
- 14. Wang, Z.Q. Algorithm Research and System Implementation of Underwater Target Detection and Tracking Based on Forward-Looking Sonar; Harbin Engineering University: Harbin, China, 2019.
- 15. Li, J. Research on Target Detection and Tracking of Forward-Looking Sonar; Harbin Engineering University: Harbin, China, 2019.
- 16. Li, H.G. Study on Moment Technique Applications in Underwater Acoustic Image Classification and Recognition; Dalian University of Technology: Dalian, China, 2018.
- 17. Kim, W.Y.; Kim, Y.S. A region-based shape descriptor using Zernike moments. *Signal Process Image Commun.* **2020**, *16*, 95–102. [CrossRef]
- 18. Hu, M.K. Visual pattern recognition by moment invariants. IRE Trans. Inf. Theory 1962, 8, 179–187.
- 19. Teague, M.R. Image analysis via the general theory of moments. J. Opt. Soc. Am. 1980, 70, 1468. [CrossRef]
- 20. Mukundan, R.; Ramakrishnan, K.R. Fast computation of Legendre and Zernike moments. *Pattern Recogn.* **1995**, *28*, 1433–1442. [CrossRef]
- Carpenter, J.; Clifford, P.; Fearnhead, P. Improved particle filter for non-linear problems. *IEE Proc.-Radar Sonar Navig.* 1999, 146, 2–7. [CrossRef]

- 22. Wang, F. Particle filters for visual tracking. Commun. Comput. Inf. Sci. 2011, 152, 107–112.
- 23. Mikami, D.; Otsuka, K.; Yamato, J. Memory-based particle filter for tracking objects with large variation in pose and appearance. In Proceedings of the European Conference on Computer Vision, Heraklion, Greece, 5–11 September 2010; pp. 215–228.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.