

Research on Adaptive Regional Classification and Prediction for Underwater Navigation Based on Genetic Algorithm and Support Vector Machine Model

Wenjie Zhang^{1, a}

¹School of Surveying and Land Information Engineering, Henan Polytechnic University, Jiaozuo 454003, China

^a1723980263@qq.com

Abstract: This paper primarily investigates the selection of matching regions in a gravity matching navigation system for submarines. When employing Support Vector Machines (SVM) for adaptive region classification, traditional SVMs necessitate manual adjustment of hyperparameters according to different scenarios. To address this limitation, our work integrates Genetic Algorithm (GA) with SVM for the selection of suitable matching regions in underwater gravity-assisted navigation. The experimental results demonstrate that this approach can provide valuable references for the division of adaptive regions and the design of navigation routes for underwater vehicles.

Keywords: Gravity matching navigation, Support vector Machine, Suitable area.

1. Introduction

As global attention on the development and utilization of marine resources grows, the navigation and positioning capabilities of underwater vehicles have emerged as a critical area of scientific inquiry[1]. These vehicles primarily rely on Inertial Navigation Systems (INS), which are self-contained, stealthy, and offer high short-term accuracy. Despite these advantages, INS suffers from errors accumulating over time due to factors like gyro drift and accelerometer bias, making long-term mission accuracy challenging without external corrections [2]. The stable and well-distributed temporal-spatial characteristics of marine gravity fields present an effective solution for correcting these INS inaccuracies[3]. By utilizing onboard gravity sensors, underwater vehicles can gather gravity field data during their journey. This data is then compared against a preloaded reference map database, enabling the identification of the most accurate matching position through specialized algorithms. The correction of inertial navigation errors using Earth's gravity field not only enhances navigation precision but also opens new avenues in the autonomous navigation of underwater vehicles. This method represents a promising research trend, aiming to leverage natural environmental features for improved underwater navigation reliability and efficiency. This approach underscores the potential for integrating environmental mapping with traditional navigation systems to achieve more reliable long-duration underwater missions[4].

2. Constructing a Support Vector Machine Classifier

2.1. Gravity field characteristic parameters

Gravity field characteristics can be described from various aspects such as statistics, correlation, and frequency [5].

(1) Standard deviation of gravity anomalies, the calculation formula is as follows:

$$\sigma = \sqrt{\frac{1}{mn-1} \sum_{i=1}^m \sum_{j=1}^n [g(i, j) - \bar{g}]^2} \quad (1)$$

Where $\bar{g} = \frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n g(i, j)$ is mean gravity anomaly,

$g(i, j)$ is the value at position (i, j) in the gravity anomaly map.

(2) Standard deviation of slope, the calculation formula is as follows:

$$\sigma_s = \sqrt{\frac{1}{mn-1} \sum_{i=1}^m \sum_{j=1}^n [S(i, j) - \bar{S}]^2} \quad (2)$$

Where $\bar{S} = \frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n S(i, j)$, $S = \arctan(\sqrt{S_\phi + S_\lambda})$,

$$S_\lambda = \frac{1}{2 * scGrid} [g(i, j+1) + g(i+1, j+1) - g(i, j) - g(i+1, j)],$$

$$S_\phi = \frac{1}{2 * scGrid} [g(i+1, j) + g(i+1, j+1) - g(i, j) - g(i, j+1)].$$

(3) Roughness, the calculation formula is as follows:

$$R = (R_\lambda + R_\phi) / 2 \quad (3)$$

Where $R_\phi = \frac{1}{n(m-1)} \sum_{i=1}^{m-1} \sum_{j=1}^n |g(i, j) - g(i+1, j)|$,

$$R_\lambda = \frac{1}{m(n-1)} \sum_{i=1}^m \sum_{j=1}^{n-1} |g(i, j) - g(i, j+1)|.$$

(4) Gravity anomaly difference entropy, the calculation formula is as follows:

$$H = -\sum_{i=1}^m \sum_{j=1}^n P_{ij} \log_2 P_{ij} \quad (4)$$

$$\text{Where } P_{ij} = \frac{D_{ij}}{\sum_{i=1}^m \sum_{j=1}^n D_{ij}}, D_{ij} = \frac{|g(i, j) - \bar{g}|}{g}$$

(5) Correlation coefficient, the calculation formula is as follows:

$$P = (P_\lambda + P_\phi) / 2 \quad (5)$$

$$P_\lambda = \frac{1}{m\sigma(n-1)} \sum_{i=1}^m \sum_{j=1}^{n-1} |g(i, j) - \bar{g}| |g(i, j+1) - \bar{g}| \quad (6)$$

$$P_\phi = \frac{1}{n\sigma(m-1)} \sum_{i=1}^{m-1} \sum_{j=1}^n |g(i, j) - \bar{g}| |g(i, j+1) - \bar{g}| \quad (7)$$

2.2. Particle Swarm Optimization Support Vector Machine

2.2.1. Support Vector Machine Principles

Support Vector Machine (SVM) is a supervised learning model used for classification and regression analysis[6]. The core idea of SVM is to find a hyperplane in a high-dimensional space that separates data points of different classes as much as possible while maximizing the margin between the classes. Additionally, by employing what is known as the kernel trick, SVM can efficiently perform non-linear classification.

$$\begin{cases} \max \theta(\alpha) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i,j=1}^N Y_i Y_j \alpha_i \alpha_j \varphi(X_i)^T \varphi(X_j) \\ s.t. \sum_{i=1}^N \alpha_i Y_i = 0 (i=1 \sim N), \alpha_i \geq 0 \end{cases} \quad (8)$$

The Gaussian Kernel Support Vector Machine (RBF-SVM) is a type of Support Vector Machine model that uses the Radial Basis Function (RBF) as its kernel. The RBF kernel, also known as the Gaussian kernel, is one of the most commonly used kernels in SVM, especially suitable for non-linear classification problems. It works by mapping the original input space to a higher-dimensional space, making it possible to find a hyperplane that can linearly separate the data points in this high-dimensional space [7]. The mathematical expression for the Gaussian kernel function is:

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \quad (9)$$

This paper adopts the radial basis function kernel, bring it into formula (7):The classification function of the optimal hyperplane is expressed as:

$$f(x) = \text{sgn}\left[\sum_{i=1}^m \alpha_i Y_i K(x_i, x_j) + b\right] \quad (10)$$

In the context of RBF-SVM, selecting the appropriate penalty parameter C and Gaussian kernel parameter g is vital for enhancing classification performance. Consequently, to improve classification outcomes, standard Particle Swarm Optimization (PSO) is utilized to optimize these hyperparameters.

2.2.2. Genetic Algorithm

Genetic Algorithm (GA) is a search heuristic that mimics the process of natural selection and genetic mechanisms, belonging to the category of evolutionary algorithms[8]. It draws inspiration from Darwin's theory of natural selection, particularly the principle of "survival of the fittest," and addresses problems by simulating operations found in biological evolution such as inheritance, mutation, selection, crossover, and recombination[9]. The formula is as follows:

$$xi = (di1, di2 \dots diD), i = 1, 2 \dots NP \quad (11)$$

For each individual in the population, the mutation vector in differential evolution is generated in the following way:

$$v_i = x_{r1} + F(x_{r2} - x_{r3}) \quad (12)$$

Where r1r1, r2r2, and r3r3 are randomly selected from the population such that r1≠r2≠r3, and F is the scaling factor. Typically, the scaling factor F is chosen within the range [0, 2].

Below is the crossover formula for genetic algorithms:

$$u_{i,j} = \begin{cases} v_{i,j}, & \text{if } \text{rand} < \text{CR} \text{ and } \text{rand}(1, D) = j \\ x_{i,j}, & \text{if } \text{rand} < \text{CR} \text{ and } \text{rand}(1, D) \neq j \end{cases} \quad (13)$$

Below is the method for selecting individuals for the next generation:

$$x_i = \begin{cases} u_i, & \text{if } \text{fit}(u_i) \text{ superior to } \text{fit}(x_i) \\ x_i, & \text{else} \end{cases} \quad (14)$$

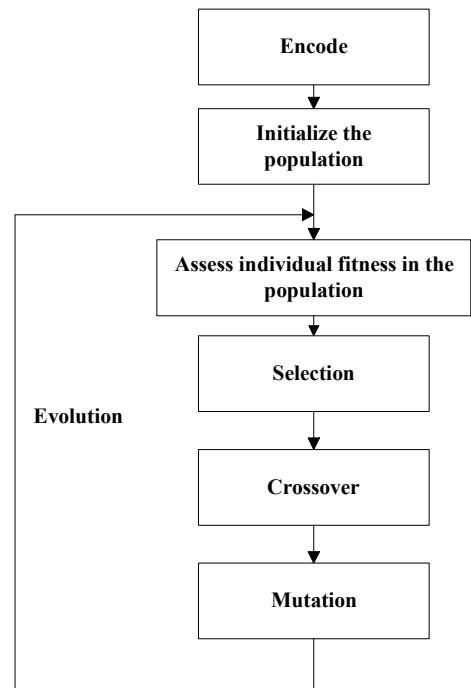


Figure 1. Flowchart of Genetic Algorithm

3. Results

This study utilized gravity anomaly data from the University of California, San Diego website (<http://topex.ucsd.edu/>), with a resolution of $1' \times 1'$. The interpolated image has a spatial resolution of approximately $100 \text{ m} \times 100 \text{ m}$. The window size was set at 150×150 grid cells, with a sliding step of 150 grid cells. Each region thus covers an actual area of approximately $15 \text{ km} \times 15 \text{ km}$. This paper divides the experimental area into 196 sub-regions and conducts simulation experiments using the TERCOM algorithm. Using average positioning accuracy as the criterion, regions with an accuracy of $100\sqrt{2}$ meters or less are

designated as suitable areas, while all other regions are designated as unsuitable areas. The simulation parameters are set as follows: the constant drift of the gyroscope in the inertial navigation system is set to $0.05^\circ/\text{h}$, and the constant error of the accelerometer is set to 0.04 m/s^2 . The simulated speed is 10 m/s . The number of samples is 80, with a sampling period of 10 seconds. A random error with a standard deviation of 1 mGal is superimposed on the sampled values as real-time measurement data for the gravimeter, and the TERCOM algorithm is used for trajectory matching. The distribution of positioning accuracy obtained from the simulation experiment is shown in Figure 2.

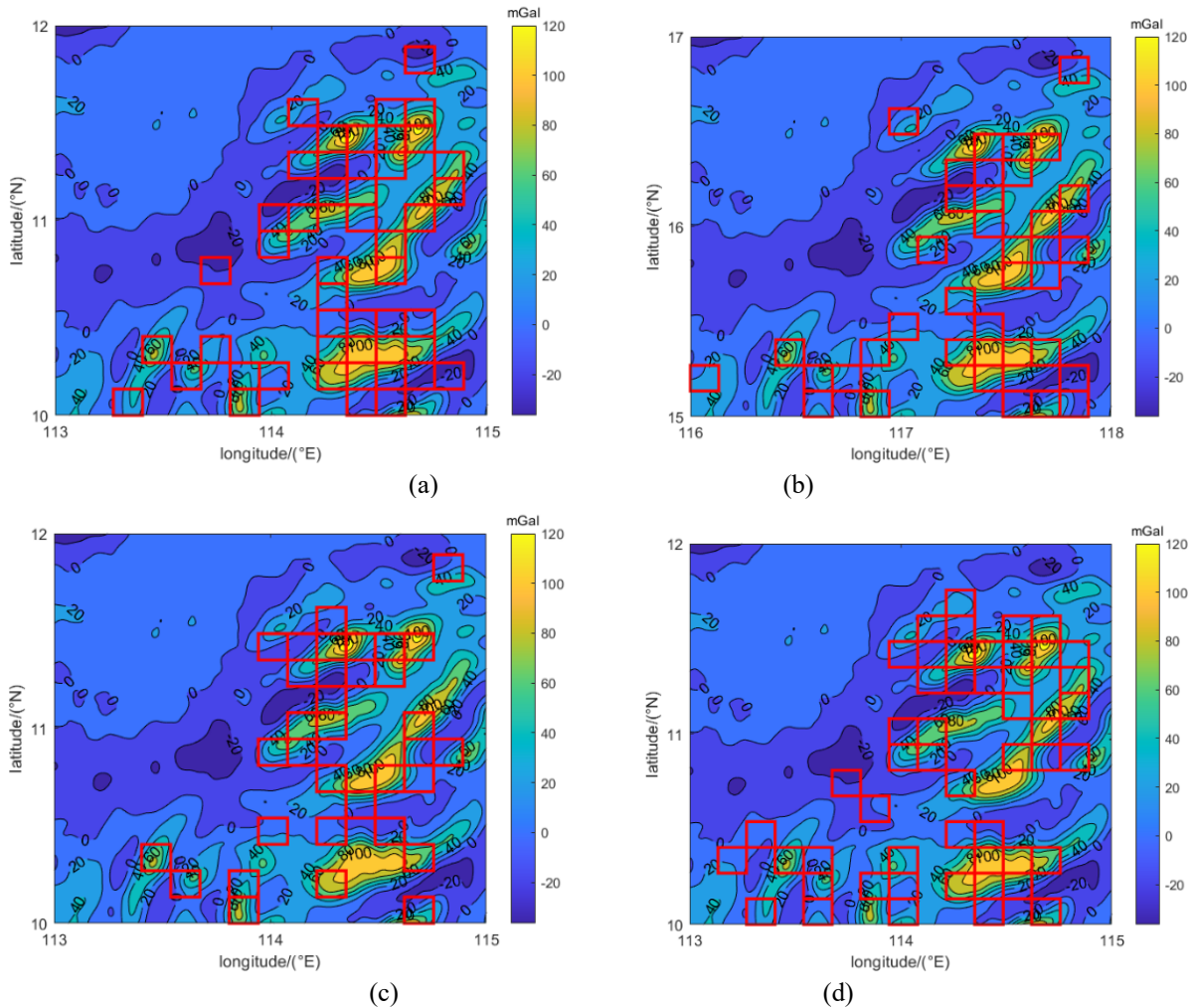


Figure 2. Calibration of Suitable areas. (a) Pre-calibrated Suitable areas in the east-west direction; (b) Pre-calibrated Suitable areas in the northeast direction; (c) Pre-calibrated Suitable areas in the north-south direction; (d) Pre-calibrated Suitable areas in the northwest direction

The dataset is divided into a training set, a test set, and a validation set in a ratio of 6:2:2. The SVM algorithm uses the LIBSVM tool developed by Lin et al., with the code compilation environment being MATLAB R2021a[10]. The standard Genetic Algorithm (GA) is used to optimize the Support Vector Machine, with the following parameter settings: penalty parameter ranges from $[0.1, 100]$; kernel function parameter ranges from $[0.00001, 10]$. The parameter settings for the Genetic Algorithm are as follows: the population size is set to 100; the number of iterations is set to 500 generations; the crossover probability is set to 0.9; the

mutation probability is set to 0.01; an elitism strategy is employed, retaining the top 2 individuals directly into the next generation to ensure that the best solutions found are not lost, which helps maintain and improve the overall quality of the population.

This indicates that for each direction, specific values have been set for the kernel function parameter and the penalty parameter to optimize the performance of the SVM model

Table 1. Performance results of various classifiers on the test set.

Direction	accuracy
East-west heading	92%
North-south heading	91%
Northeast heading	93%
Northwest heading	89%

Manually adjusting these parameters is time-consuming and challenging, whereas Genetic Algorithms (GA) can automatically search for the best parameter combinations, thereby saving time and effort. GA employs a search strategy based on evolutionary principles, allowing for extensive searches across the entire solution space. This capability helps in finding the global optimum or a solution close to the global optimum, thus avoiding the problem of getting trapped in local optima, which is common with traditional optimization methods. As shown in Table 1, the classifier's accuracy reaches above 90% in all directions except for the northwest, where it stands at 87%. Experiments have shown that the method proposed in this paper can more comprehensively divide the adaptation regions of the study area.

4. Conclusions

This paper primarily investigates underwater navigation technology based on gravity-adaptive regions, utilizing Support Vector Machine (SVM) classification methods for gravity-assisted navigation. By incorporating kernel functions, SVM can accurately capture nonlinear relationships and complex dynamic changes within the data, offering excellent adaptability and robustness. To further enhance the classifier's accuracy, this study introduces Genetic Algorithm (GA) to optimize the SVM. The experiment proves that the method proposed in this paper can divide the adaptation region of the study area more comprehensively.

References

- [1] Zou,J.S, Xiao,y, Sun,A.B, Wei,J.C, Meng,N, and Ai,S.J, "Research on relationship between gravity anomaly feature quality and matching navigation algorithm," Science of Surveying and Mapping, vol. 45, no. 10, pp. 33–40, 2020, doi: 10.16251/j.cnki.1009-2307.2020.10.006.
- [2] Chen,K.N, Xiao,Y, Zhang,B.J, Cao,J, Wang,Y.K, and Hong,X.D, "Comparison of Adaptive Evaluation Methods of Underwater Gravity Matching Navigation," Journal of Geodesy and Geodynamics, pp. 1–12, doi: 10.14075/j.jgg.2023.10.176.
- [3] Z. Wang, Y. Huang, M. Wang, J. Wu, and Y. Zhang, "A Computationally Efficient Outlier-Robust Cubature Kalman Filter for Underwater Gravity Matching Navigation," IEEE Trans. Instrum. Meas., vol. 71, pp. 1–18, 2022, doi: 10.1109/TIM.2022.3141153.
- [4] W. Ai, S. Zhong, Y. Leng, and S. Wang, "A Real-Time Autonomous Gravity Measurement Method for Underwater Gravity-Aided Navigation," IEEE Access, vol. 11, pp. 135728–135737, 2023, doi: 10.1109/ACCESS.2023.3338537.
- [5] Xiao,Y, Zhang,Y.J, Cao,J, Chen,K.N, Wang,Y.K, and Hong,X.D, "Suitability Analysis of Gravity Matching Navigation Based on Multiple Attribute Decision Making Theory," Geomatics and Information Science of Wuhan University, vol. 48, no. 7, pp. 1089–1099, 2023, doi: 10.13203/j.whugis20230073.
- [6] M. A. Ganaie and M. Tanveer, "KNN weighted reduced universum twin SVM for class imbalance learning," Knowledge-Based Systems, vol. 245, p. 108578, Jun. 2022, doi: 10.1016/j.knsys.2022.108578.
- [7] V. Ranaee, A. Ebrahimzadeh, and R. Ghaderi, "Application of the PSO–SVM model for recognition of control chart patterns," ISA Transactions, vol. 49, no. 4, pp. 577–586, Oct. 2010, doi: 10.1016/j.isatra.2010.06.005.
- [8] G. Acampora, A. Chiatto, and A. Vitiello, "Genetic algorithms as classical optimizer for the Quantum Approximate Optimization Algorithm," 2023.
- [9] B. Alhijawi, "Genetic algorithms: theory, genetic operators, solutions, and applications".
- [10] V. Vapnik, S. E. Golowich, and A. J. Smola, "Support Vector Method for Function Approximation, Regression Estimation and Signal Processing".