THE JOURNAL OF NAVIGATION (2017), **70**, 82–100.
doi:10.1017/S0373463316000369

Terrain Matching Positioning Method Based on Node Multi-information Fusion

Ye Li, Rupeng Wang, Pengyun Chen, Peng Shen and Yanqing Jiang

(*Science and Technology on Underwater Vehicles laboratory, Harbin Engineering University, 150001, China*)
(E-mail: wangrupeng@hrbeu.edu.cn)

Measurement bias and lack of terrain features often cause false peaks during underwater terrain matching positioning, that is, there is more than one peak near the real position. Previous methods to address this problem have increased the number of measurement beams, but this also increases the data processing time and energy consumption. At the same time, the ratio of measured information that is used does not increase. In other words, we should increase the ratio of measured information that is used, not simply increase the amount of information that is measured. Conventional matching algorithms only use the height of nodes without considering surface information, which is composed of height and the position of multiple nodes in three-dimensional space. Multi-beam sonar can obtain the three-dimensional distribution of terrain nodes. This node information is not just a height sequence, as it is used in previous methods. If we consider the nodes as a three-dimensional distribution of points with height and position information, this increases the matching position information and more of the terrain features can be extracted from the same measured data. Hence, in this paper, a terrain positioning method called the Node Multi-information Fusion (NMIF) is presented. This method focuses on improving the stability and accuracy degraded by bias in the Digital Elevation Map (DEM), terrain repeatability, and other factors. First, the concept of a Single Node Data Packet (SNDP) is introduced. The SNDP includes elevation and surface information surrounding the node, such as roughness, gradient, and slope. This additional topographic feature information improves the robustness and accuracy of the system. A computer simulation using actual ocean bottom topography verifies the advantages of the proposed NMIF algorithm.

KEYWORDS

1. Autonomous Underwater Vehicle (AUV). 2. Terrain reference navigation.
3. Node data packet. 4. Multi-information fusion.

Submitted: 4 May 2015. Accepted: 26 May 2016. First published online: 8 July 2016.

1. INTRODUCTION.

An Autonomous Underwater Vehicle (AUV) is an indispensable piece of engineering equipment in ocean exploration. Because of the complexity of the underwater environment, the research and development of AUVs has

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press

encountered many difficulties, and precise navigation is one of them. At present, underwater navigation is mainly based on the Inertial Navigation System (INS), but the cumulative error of INS is an unsolvable problem. Regularly breaking the surface to "zero out" the position error greatly increases energy consumption, and for long-endurance AUVs, this is particularly undesirable. Terrain positioning is therefore an important problem. To solve this problem, researchers have proposed terrain-aided navigation methods that were first used in the field of aviation. Initial studies focused on improvements based on land terrain matching navigation systems. There are already a number of countries and institutes that have developed their own underwater terrain matching navigation system.

For instance, the Norwegian Defense Research Establishment developed the TerrLab simulation system (Hagen, 2006) and have conducted several sea trials based on the HUGIN AUV. In 2010, they conducted a sea trial for five hours in Oslo, and the navigation error was only 5 m (Anonsen and Hagen, 2010; Hagen et al., 2010). Stanford University and the Monterey Bay Aquarium Research Institute developed a low-cost terrain navigation method that is suitable for long-range AUVs. They took an MBARI Dorado AUV out into Monterey Bay in 2008 to verify the feasibility of using low-precision measuring equipment for terrain-aided navigation and used low-accuracy stern projections and a terrain navigation filter, achieving an accuracy of 4–10 m (Meduna et al., 2008; 2010). The Swedish Royal Institute of Technology (Nygren, 2008), University of Southampton (Carreno et al., 2010), and University of Tokyo with the Japan Institute of Marine Engineering (Nakatani et al., 2009) have also developed their own underwater terrain matching navigation systems. In addition, Peking University (Zhang et al., 2003), Wuhan University (Zhang et al., 2011; 2014), and Harbin Engineering University (Chen, 2013; Tian, 2007) have also studied underwater terrain matching navigation systems. However, most studies are based on improving the aviation terrain matching algorithm or single beam terrain matching system. Related studies include the improved TERCOM algorithm (Liu, 2003) and Image Matching Technology (Zhang et al., 2011).

In contrast to land-based environments, the underwater environment makes it difficult for current bathymetric technology to obtain a terrain map that is as precise as a land-based map. In addition, the underwater terrain is flatter and AUVs move slowly, which decrease the signal-to-noise ratio and reduce positioning accuracy, reliability, and stability. This problem is particularly prominent in flat regions (Chen, 2013; Xie, 2005). In this paper, the Node Multi-Information Fusion (NMIF) algorithm is proposed, which effectively improves the accuracy of flat terrain positioning and measurement stability. This method was verified using a real underwater terrain map.

2. CONSTRUCTION OF NODE DATA PACKETS.

In terrain matching, positioning methods are based on the Digital Elevation Map (DEM), which stores the height of each grid node sampled from the ocean floor. In the Measured Terrain Map (MTM), all heights form a sequence of height values. Matching algorithms determine the position of a point in the DEM by matching an MTM height sequence with the height sequence from the DEM. This point is called a positioning point, but it is never the actual position of the AUV. In a flat region, false position peaks caused by low numbers of measured terrain nodes and limited feature information are inevitable

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press

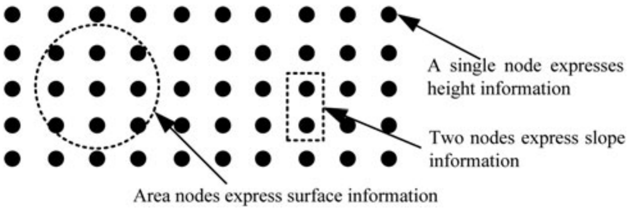

A single node expresses height information
Two nodes express slope information
Area nodes express surface information

Figure 1. Information contained in single and multiple nodes.

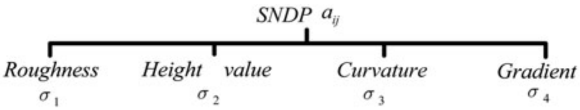


Figure 2. Construction of an SNDP.

(Chen, 2013; Xie, 2005; Nygren, 2005), so the focus of this study is to extract more information from limited topographic terrain features. In fact, multi-beam methods increase the number of nodes measured per measurement period, and this increases the amount of position information, which does improve the accuracy and efficiency of terrain matching. However, there is no improvement in the utilisation of node information. There is a difference between a terrain profile formed from a single beam and a topographic profile formed from multiple beams. The surface includes significantly more topographic features than a terrain profile does, and it does not significantly improve accuracy and efficiency to only use height information. A single node includes only height information, but if it is observed using a multi-beam MTM, it is possible to add surrounding surface information into a node data packet, causing it to become more distinct. The construction of such a data packet, the Single Node Data Packet (SNDP), is shown in Figure 1.

In order to make full use of the node information, this paper proposes a new terrain matching method based on the SNDPs. Suppose a_{ij} denotes the data packet of node (i, j) in the DEM, expressed as an ξ-dimensional vector, $a_{ij} = [\sigma_1, \sigma_2 \cdots \sigma_\xi]$.

In Figure 2 an SNDP contains the information surrounding one node and consists of four information units. How much information is included in these units is decided by many factors, but the reliability of the packet information must be ensured. In contrast to a single node, which only contains height information, more information is included in the SNDP. Hence, one can think of the SNDP as an expansion of the node information, or a fusion of multiple types of information regarding a certain single node.

3. TERRAIN MATCHING BASED ON PACKETS.

Similarity is the key to positioning, and a packet contains a variety of information with which to calculate it. We assume that every item of information is independent of each other. A sub-element matching area is a part of the search area, and its location and size is provided by inertial or reckoning navigation and the corresponding navigation error. The search area, sub-element matching area, node, and surrounding node information are shown in Figure 3.

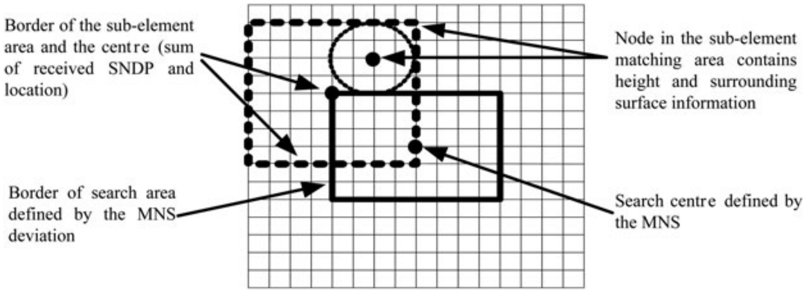

Border of the sub-element area and the centre (sum of received SNDP and location)
Node in the sub-element matching area contains height and surrounding surface information
Border of search area defined by the MNS deviation
Search centre defined by the MNS

Figure 3. Search area, search centre, and sub-element matching search area in the DEM.

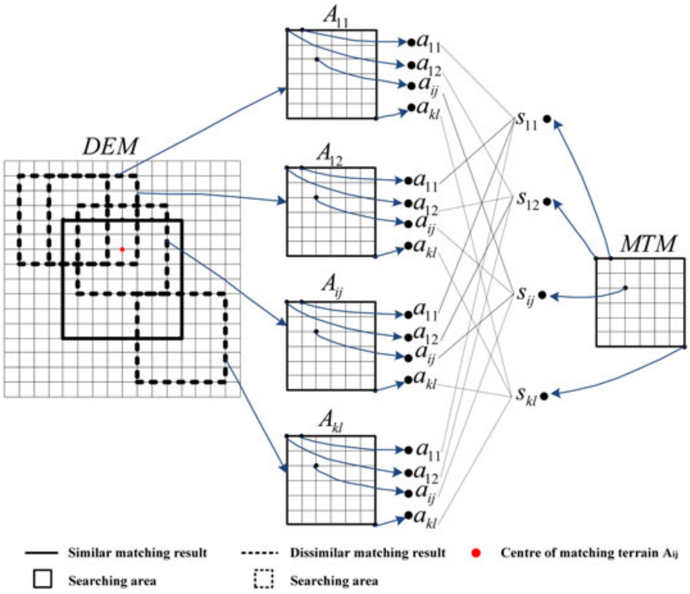
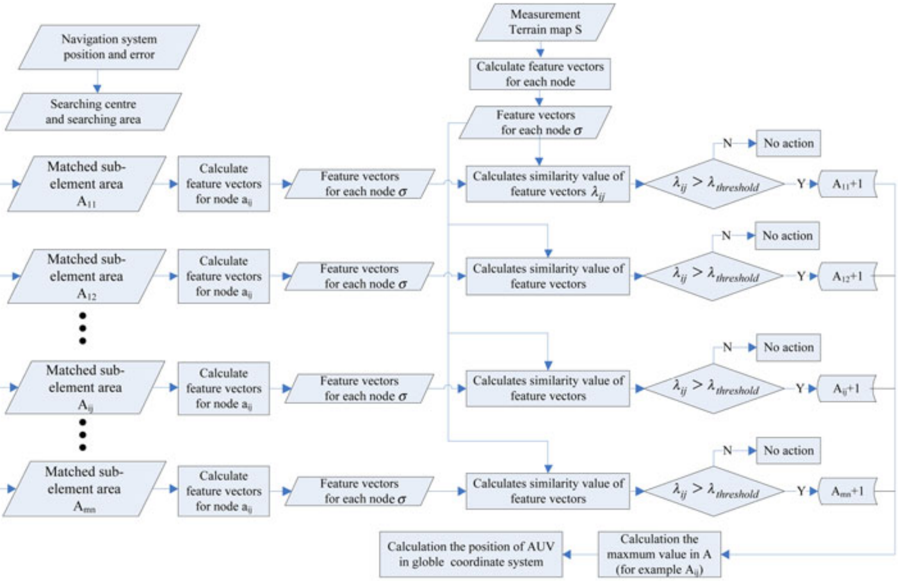


Figure 4. Obtaining and comparing the SNDPs from the DEM and MTM.

3.1. *Matching SNDPs.* The MTM contains SNDPs to be matched with the corresponding nodes in the DEM, and the matching algorithm decides which SNDP in the DEM is most similar to the SNDP from the MTM. For example, as shown in Figure 4, suppose s_{ij} is a node in the DEM and there are $k \times l$ corresponding nodes from the MTM that must be compared with it. Further suppose that the matching algorithm considers node a_{ij} from sub-element areas A_{11} and A_{ij} to be similar to s_{ij}. In this case, A_{11} and A_{ij} have SDNP s_{ij} added to them. Figure 4 shows the process that determines the SNDPs to be matched in the DEM and MTM and the corresponding comparison of the SNDPs. If an SNDP in the DEM qualifies as similar, as defined in the algorithm, then the node in the DEM has that SNDP added to it. The centre of the matched sub-element search area in the DEM records the number of added SNDPs, and a higher number indicates greater similarity. Of course, if a node in a certain

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press


Navigation system position and error
Searching centre and searching area
Measurement Terrain map S
Calculate feature vectors for each node
Feature vectors for each node σ
Matched sub-element area A11
Calculate feature vectors for node aij
Feature vectors for each node σ
Calculates similarity value of feature vectors λij
λij > λthreshold
N
No action
Y
A11+1
Matched sub-element area A12
Calculates similarity value of feature vectors
A12+1
Matched sub-element area Aij
Aij+1
Matched sub-element area Amn
Amn+1
Calculation the position of AUV in globle coordinate system
Calculation the maxmum value in A (for example Aij)

Parameters σ, λ_{ij}, and a_{ij} are parameters from A. Their values differ for each A.

S: MTM A: Sub-element area for position search

Figure 5. Flowchart of the NMIF algorithm.

sub-element in the matched area meets the similarity condition, it can always add node s_{ij}. In Figure 4, s_{11} is similar to SNDP α_{11} in sub-element areas A_{11}, A_{12}, and A_{ij}, so A_{11}, A_{12}, and A_{ij} will have one SNDP added each. As shown in Figure 4, A_{11} adds one SNDP, A_{12} adds one SNDP, A_{ij} adds three SNDPs, and A_{kl} adds no SNDPs. In this case, sub-element area A_{11} is the most similar to the MTM, and the searching point represented by A_{ij} is marked by a red dot.

3.2. *Algorithm implementation.* The number of SNDPs added at a certain sub-element area in the DEM is a measure of the level of similarity between the MTM and the sub-element search area. In Figure 5, it can be seen that only the similarity between the SNDPs from the sub-element search area and the SNDPs from the MTM exceed the threshold defined in the algorithm. The node in the sub-element search area adds the packet, and the value of A_{ij} is incremented. At this point, A_{ij} represents both the added data packets and the centre of the sub-element search area. Details of the procedure are as follows.

3.2.1. *Node similarity measurement -Algorithm.* (1) For example, if there is an $m \times n$ node DEM (m and n represent the number of nodes in the x and y directions, respectively), the feature vector of each node must be calculated, so the algorithm must calculate $m \times n$ feature vectors $(\sigma_1, \sigma_2 \ldots \sigma_\xi)$, where ξ represents the number of features of each node. The algorithm marks the centre of the sub-element search area with A_{ij} $(i = 1, 2, \ldots m, j = 1, 2, \ldots n)$. Assuming that every sub-element search area contains $k \times l$ nodes and each feature vector has ξ dimensions, the algorithm marks the sub-element search area with a_{ij} $i = 1, 2, \ldots k$, $j = 1, 2 \ldots l$, and $a_{ij} = (\sigma_1, \sigma_2, \ldots \sigma_\xi)$.

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press

Different nodes will have different values for $\sigma_1\ \sigma_2\ \sigma_3 \ldots \sigma_\xi$. The relation between A_{ij}, a_{ij}, and $(\sigma_1,\ \sigma_2 \ldots \sigma_\xi)$ is shown below.

$$\begin{bmatrix} A_{11} & A_{12} & \ldots & A_{1n} \\ A_{21} & A_{22} & \ldots & A_{2n} \\ \vdots & \vdots & A_{ij} \longrightarrow & \vdots \\ A_{m1} & A_{m2} & \ldots & A_{mn} \end{bmatrix} \longrightarrow \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1l} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & a_{ij} \longrightarrow & \vdots \\ a_{k1} & a_{k2} & \ldots & a_{kl} \end{bmatrix} \longrightarrow \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \vdots \\ \sigma_\xi \end{pmatrix} \tag{1}$$

The MTM is processed in the same way to obtain its feature vector. It is marked with $(\sigma_1,\ \sigma_2,\ \sigma_3 \cdots \sigma_\xi)$ because a MTM has only one unit S. Hence, the relation between $(\sigma_1,\ \sigma_2,\ \sigma_3 \cdots \sigma_\xi)$ and S is as shown below.

$$S = \begin{bmatrix} s_{11} & s_{12} & \ldots & s_{1l} \\ s_{21} & s_{22} & \ldots & s_{2l} \\ \vdots & \vdots & s_{ij} \longrightarrow & \vdots \\ s_{k1} & s_{k2} & \ldots & s_{kl} \end{bmatrix} \longrightarrow \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \vdots \\ \sigma_\xi \end{pmatrix} \tag{2}$$

(2) Using a similarity decision function, the algorithm calculates the similarity between two feature vectors. For example, given a node $s(i, j)$ from MTM S and node $a(i, j)$ from sub-element search area A_{ij} $i = 1, 2, \ldots m$, $j = 1, 2, \ldots n$, which is the node corresponding to $s(i, j)$, the algorithm sets λ_{ij} $i = 1, 2, \ldots m$, $j = 1, 2, \ldots n$, to be the similarity measurement between $a(i, j)$ and $s(i, j)$, where $\lambda_{threshold}$ represents the similarity threshold. That is, if the similarity $\lambda_{ij} > \lambda_{threshold}$, $s(i, j)$ and $a(i, j)$ are considered to represent the same node. At the same time, sub-element search area A_{ij} adds an SNDP and its value is incremented by one.

(3) Step 2 is repeated until all nodes have been compared with the corresponding node. Finally, the algorithm calculates the sum of all the SNDPs added to sub-element search area A_{ij} as follows.

$$A_{ij} = \sum_{i=1}^{m} \sum_{j=1}^{n} \delta_{ij} \tag{3}$$

The value of δ_{ij} is defined by the following formula:

$$\delta_{ij} = \begin{cases} 1 & \lambda_{ij} \geq \lambda_{threshold} \\ 0 & \lambda_{ij} < \lambda_{threshold} \end{cases} \tag{4}$$

The algorithm next determines the maximum $A_{\max}$ from all A_{ij} ($i = 1, 2, \ldots m$, $j = 1, 2, \ldots n$). Here, A_{ij} represents the centre of the sub-element search area and simultaneously the sum of the SNDPs that sub-element search area A_{ij} has added. Hence, A_{ij} contains two elements, one is a coordinate that represents the centre of A_{ij} and another is a number that represents the sum of the SNDPs that A_{ij} has added. As mentioned, higher values of A_{ij} indicate a higher similarity between A_{ij} and S. Hence, $A_{\max}$ represents the sub-element search area that is the most similar to the real-time DEM S, and the coordinate of $A_{\max}$ is the final position.

3.2.2. *Similarity measurement function.* The similarity function is a measurement function that represents similarity (Zhang et al., 2009; Shi, 2003). We define a function that transforms the deviation between two vectors into a measurement scale λ. We use the Gaussian function to measure similarity. Suppose Θ is an n-dimensional vector, Θ_i $i = 1, 2, \ldots m$ is a sample space that contains m samples, and Θ_i also has n dimensions. We formulate Θ as follows: $\Theta = \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \vdots \\ \sigma_n \end{pmatrix}$, $\Theta_i = \begin{pmatrix} \sigma_{i1} \\ \sigma_{i2} \\ \vdots \\ \sigma_{in} \end{pmatrix}$, $n = 1, 2, \ldots$. Suppose Δ_i is the deviation between Θ and Θ_i. Using the Euclidean distance to measure the deviation between Θ and Θ_i, Δ_i can be represented by the following equation.

$$\Delta_i = \left[(\sigma_i - \sigma_{i1})^2 + (\sigma_i - \sigma_{i2})^2 + (\sigma_i - \sigma_{i3})^2 \cdots\right]^{1/2} \tag{5}$$

Our goal is now to determine the Θ_i that are similar to Θ using the value Δ_i. The problem is how to represent the level of similarity and decide whether Θ_i is similar. The solution is to use a function $f(\Delta_i)$ to transform Δ_i into a value λ_i, where $\lambda_i = f(\Delta_i)$ is the data range [0, 1], then define a threshold $\lambda_{threshold}$. If $\lambda_{ij} > \lambda_{threshold}$, then Θ_i is considered similar to Θ. From this analysis, λ should have the following properties:

1. *If* $\Delta_i = 0$ *(*$i = 1, 2, 3 \ldots$*), i.e., the deviation between* Θ *and* Θ_i *is zero,* Θ *and* Θ_i *are exactly similar, then* $\lambda_i = 1$. *If* $\Delta_i \to \infty$, Θ *and* Θ_i *are exactly dissimilar, at this point* $\lambda_i = 0$.
2. *As the similarity value increases,* λ_i *increases monotonically.*
3. *The value of* λ_i *is within the range* [0, 1].

Given the above analysis, we constructed a similarity measurement function based on the Gaussian function (Zhang and Yang, 2004). The properties of a Gaussian function mean that it can transform deviation value Δ_i into a number between [0, 1], achieving our goals. The normal form of a Gaussian function is:

$$f(x) = ae^{-(x-b)^2/c^2} \tag{6}$$

Suppose x is parameter Δ_i. At point $b = 0$, $\lambda = 1$ must be true, so $a = 1$. After reformulating the new Gaussian function by replacing $f(\Delta_i)$ with λ, we have:

$$\lambda = e^{-\Delta^2/c^2} \tag{7}$$

Parameter c is an important element that controls the shape of function λ. Hence it clearly influences the value of λ. For various c, the shape of function λ becomes slimmer as the value increases. Hence, λ is high in a small area surrounding “0”. The situation will be the reverse if parameter c is small and we decrease its value. Figure 6 shows the shape of function λ for different c, indicating the value of λ at point $\Delta = 1$.

This result shows that when c is small, the ability to measure the similarity between two vectors is significantly weakened. Hence, in practice, we should choose larger values. We discuss how to choose value c below. First, we establish the principle of similarity measurement, as shown in Figure 7.

Suppose X and Y are two measurement results at the same point. The measurement noise of the two measurements is denoted by ω_x and ω_y, respectively. Because measurements X and Y come from the same point, given our definition of similarity, we

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press

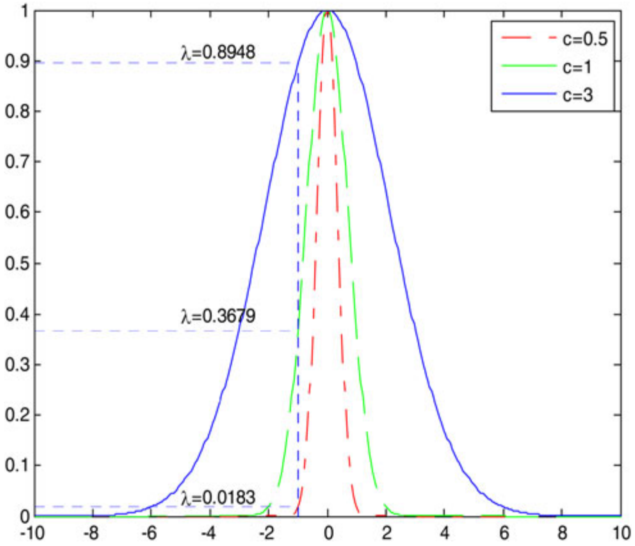


Figure 6. Value of λ when c has three different values at point $\Delta = 1$.

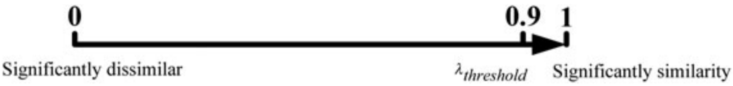


Figure 7. Similarity measurement axis.

examine the value of λ, which must exceed threshold $\lambda_{threshold}$. Suppose $\lambda_{threshold} = 0.9$. The similarity can be calculated as follows:

$$e^{-(X-Y)^2/c^2} > 0.9 \tag{8}$$

If X and Y come from the same node without measurement error, $X - Y = 0$, and clearly $\lambda = 1$. However, we must take the error into account (Chen, 2013). The similarity function is reformulated as follows:

$$e^{-(\omega_x-\omega_y)^2/c^2} > 0.9 \tag{9}$$

$$\text{Next, we calculate } c: -\frac{(\omega_x-\omega_y)^2}{c^2} > \ln 0.9 \tag{10}$$

$$\text{Then}: \quad c > \frac{|\omega_x-\omega_y|}{\sqrt{-\ln 0.9}} \tag{11}$$

$$\text{The range of values for } c \text{ is}: c > 3.081|\omega_x-\omega_y| \tag{12}$$

Usually, we assume that the measurement noise is zero-mean Gaussian white noise. As it is preferable to use a larger value, $|\omega_x - \omega_y|$ should result in a larger value, that is, $\sigma_x + \sigma_y$. Here, $\sigma_x + \sigma_y$ represents the sum of the standard deviations of the measurement noise for both measurements. We further suppose that each measurement

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press

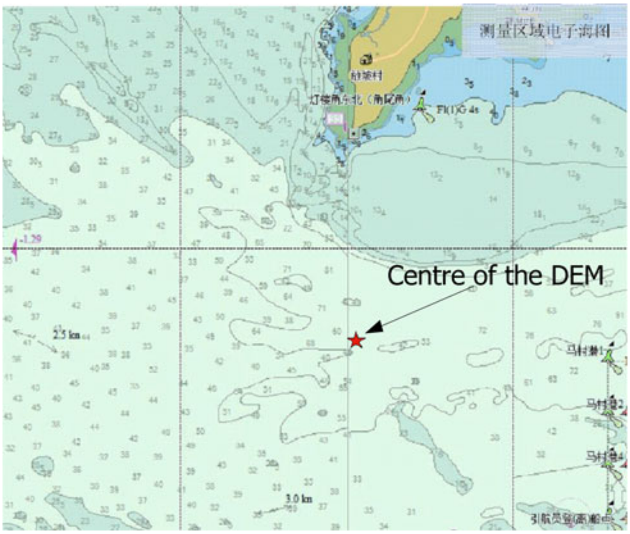


Figure 8. Electronic sea chart including the DEM area.

noise is independent. If X and Y are vectors with n dimensions, then $\sigma_x + \sigma_y$ will be a diagonal matrix with n dimensions (Xie, 2005). The above explanation describes the value of parameter c when $\lambda_{threshold} = 0.9$. If the threshold is changed, the procedure is unchanged. In order to improve stability and efficiency, choosing a higher threshold is better. Hence, we always use $\lambda_{threshold} \geq 0.9$.

4. EXPERIMENTAL ANALYSIS.

Using theoretical analysis, we obtained the structure and necessary parameters of the algorithm. We next examine the stability and robustness of the algorithm by conducting a series of experiments using an actual ocean floor DEM located in the South China Sea (Figure 8). The data was obtained using a GS+ bathymetric system and bilinear interpolation to create a DEM with a grid size of 4 × 4 m.

4.1. *Best feature selection.*

The DEM has an area of 800 × 700 m, with a grid size of 4 × 4 m. The grid size in the MTM is 2 × 2 m; hence, we used a linear interpolation algorithm to transform the DEM into a 2 × 2 m grid. A simulation MTM was obtained from the DEM and covers an area of 200 × 200 m. The centre was located at (500, 500) in the local coordinate system, the same as below. The position of nodes in the global coordinate system can be determined by the following formula:

$$\begin{cases} x = id + x_0 \\ y = jd + y_0 \end{cases} \tag{13}$$

where x_0, y_0 represent the coordinates of the origin of the local coordinates in the global coordinate system, d is the length of a grid side, and i, j represents the index of the nodes.

At the same time, we take the measurement of the circular error probability and height error into consideration. We set the circular error probability to be $\sigma_r = 0.5\,m$ and the height error probability to $\sigma = 1\,m$. They are both modelled as zero mean

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press

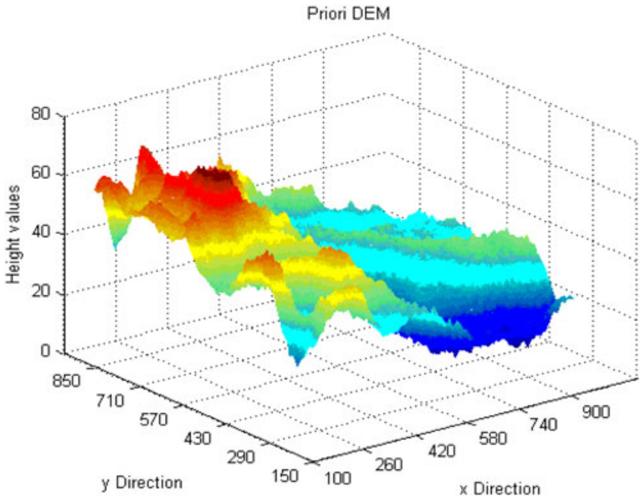


Figure 9. DEM (search area).

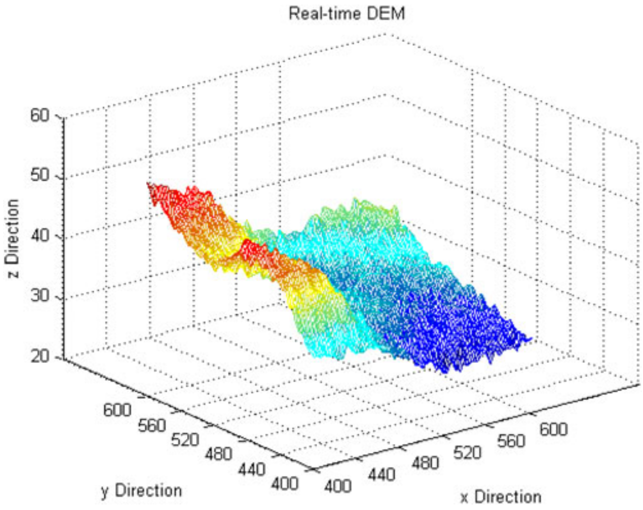


Figure 10. MTM.

Gaussian white noise. Considering the deviation in MNS, we chose an area of 200 × 200 m centred at point (500, 500). The search area and MTM are shown in Figures 9 and 10.

4.1.1. *Node packets containing height only.* The similarity measurement function is expressed as follows:

$$\lambda_{ij} = \exp\left[-\frac{1}{c^2}(\sigma_1(s) - \sigma_1(a))^2\right] \tag{14}$$

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press

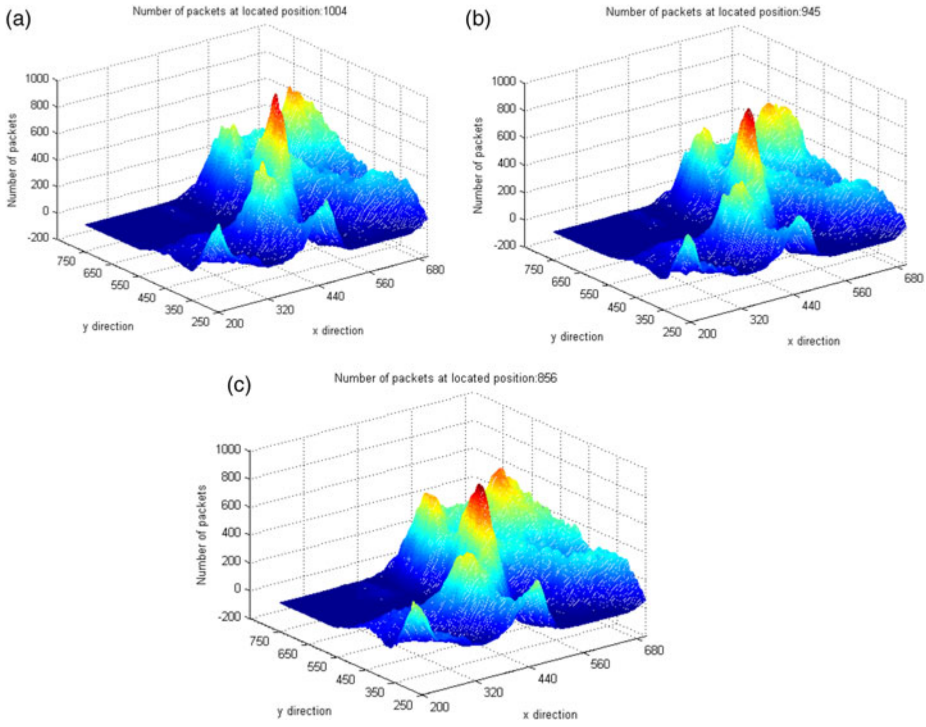


Figure 11. Distribution of the sum of SNDPs in the search area using height at various grid resolutions (a) 2 m grid (b) 4 m grid (c) 6 m grid.

In this equation, $\sigma_1(s)$ represents the height of a node in the DEM, and $\sigma_1(a)$ represents the height of a node in the MTM. We set the threshold to $\lambda_{threshold} = 0.9$ and interpolated the DEM with grids respectively equal to 2, 3, 4, 5, and 6 m into a 2 m grid. The final simulation results are shown in Figure 11 for the 2, 4, and 6 m grids. Figures 11(a)–(c) show the distribution of the sum of the SNDPs in the search area using height only. Figure 12 shows the number of SNDPs at the positioning point in the three tests. From the results, it can be seen that as the grid length increases, the number of added SNDPs surrounding the located point increases rapidly. The false peaks appear when the grid length is set to 6 m.

4.1.2. *Node packets containing surface features only.* Consider the case where there is more than one node in the DEM that is similar to the node in the MTM. In such a case, we must judge whether this node comes from the positioning area. The algorithm cannot decide by height alone because it is too similar to other positions; hence, the surface information becomes useful for distinguishing many matching nodes in a series of similar nodes. The surface information can be represented in many ways, such as by entropy information, slope, gradient, and so on. Briefly, we must choose a method that can express the features of the terrain surface effectively and accurately. In this paper, we use the sum of the height of a node and its height standard deviation in the local area that is centred on that node. The area should be extensive enough so that it can distinguish the node from all other nodes at a

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press

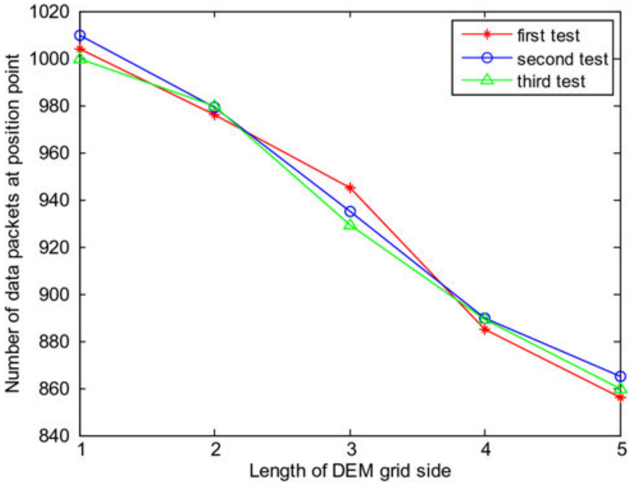


Figure 12. Sum of the SNDPs at the position point for three tests using height value as positioning information.

similar height. The feature equation is as follows:

$$\sigma_2 = \sigma_0 + \sigma_1 \tag{15}$$

where σ_2 represents the feature parameter of a node that contains height and surface information, and σ_0 represents the standard deviation of height in the local area that is centred on the matching node. In this paper, the area of the local region is $10 \times 10\ \text{m}^2$. Furthermore, σ_1 represents the height of the matching node. The similarity is measured as follows:

$$\lambda_{ij} = \exp\left[-\frac{1}{c^2}(\sigma_2(s) - \sigma_2(a))^2\right] \tag{16}$$

In Equation (16), $\sigma_2(s)$ represents the blended surface feature of the node in the MTM and $\sigma_2(a)$ represents the blended surface feature surrounding the node in the DEM. We set the threshold to $\lambda_{threshold} = 0.9$. We also chose the DEM with grids respectively equal to 2, 3, 4, 5, and 6 m interpolated into a 2 m grid. The final simulation results are shown in Figures 13(a)–(c). These figures show the distribution of the sum of SNDPs in the sub-element search area centre for the 2, 4, and 6 m grids using the surface feature only. Figure 14 shows the number of SNDPs at this position point in the three tests.

The test results show that the surface feature in the local area can identify similar nodes more easily, as the final point had more SNDPs added. On the other hand, the ability to inhibit false peaks was weakened.

4.2. *Node Multi-Information Fusion (NMIF).* From these results, it can be seen that both height and surface information have their own advantages. Both types of information converge near matches, and using surface features makes it easier to identify

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press

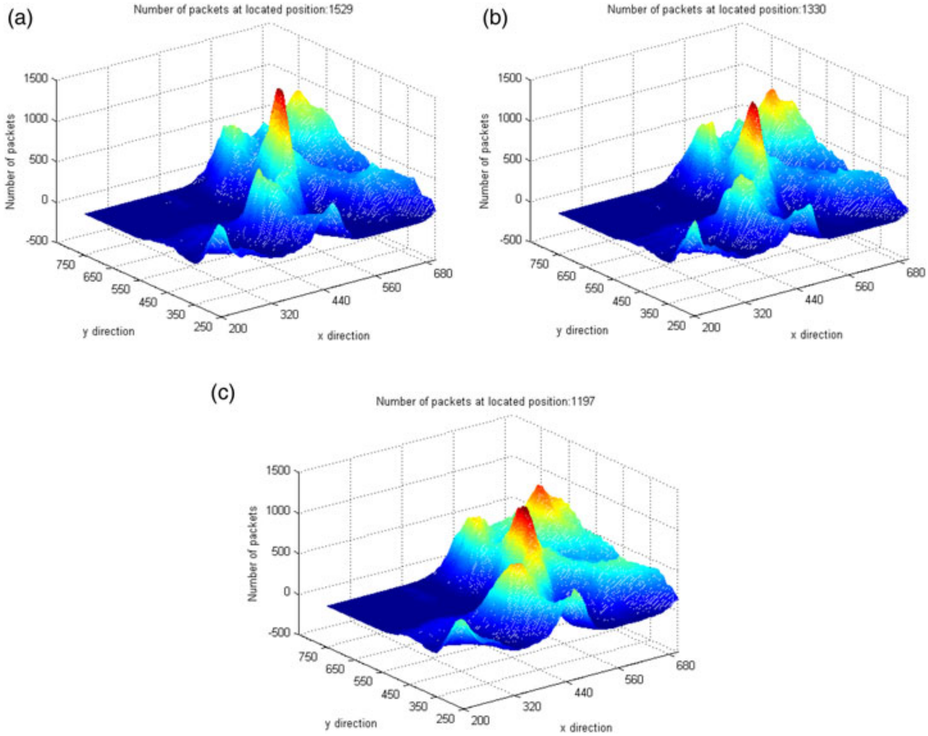


Figure 13. Distribution of the sum of SNDPs in the search area using surface features with various grid resolutions. (a) 2 m grid. (b) 4 m grid. (c) 6 m grid.

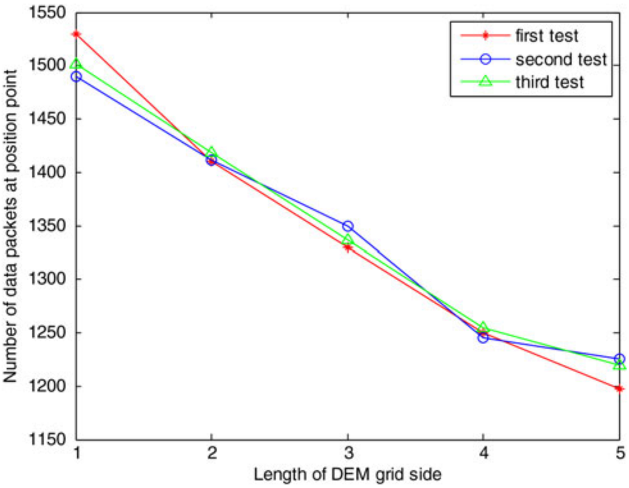


Figure 14. Sum of the SNDPs at the position point for three tests using surface features as positioning information.

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press

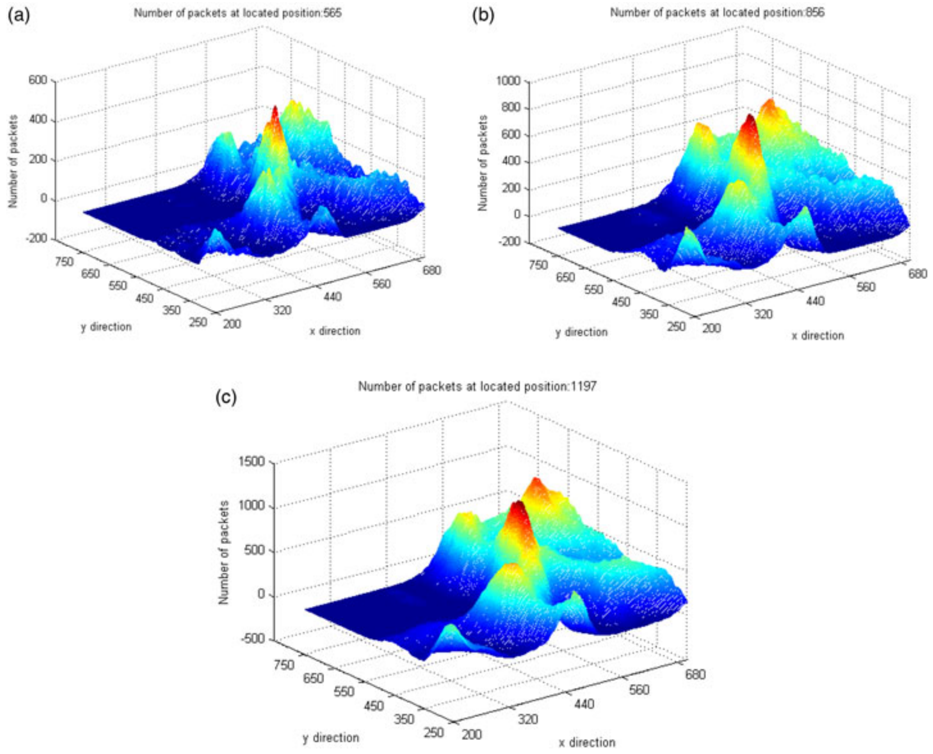


Figure 15. Distribution of the sum of SNDPs in the search area using NMIF, height value and surface features as matching information. (a) NMIF (logic fusion). (b) Height only. (c) Surface features only.

similar nodes but does not inhibit false point matches. Hence, we next fuse the two types of position information in two ways and compare them.

4.2.1. *Fusion based on logic.* For logical "and" fusion, the result is true only if all of the parameters meet the criteria. In this method, both the height and the surface height characteristics must exceed the threshold before the node in the MTM receives the SNDP.

$$\begin{cases} A_{ij} = \sum_{i=1}^{m} \sum_{j=1}^{n} \delta_{ij} \\ \delta_{ij} = \begin{cases} 1 & \lambda_{ij}^{\sigma_1} \geq \lambda_{threshold} \quad and \quad \lambda_{ij}^{\sigma_2} \geq \lambda_{threshold} \\ 0 & \qquad\qquad ealse \end{cases} \end{cases} \tag{17}$$

In this equation, $\lambda_{ij}^{\sigma_1}$ and $\lambda_{ij}^{\sigma_2}$ are at a similar scale for the height and surface features.

We next test the algorithm on the DEM 6 m grid, comparing it with the results using height only. The results are shown in Figure 15.

Figure 15(a) shows the distribution of the sum of SNDPs using logic fusion, Figure 15(b) is the result using height value only, and Figure 15(c) is the result using surface features only. This comparison shows that for the same grid precision and measurement noise, the results based on the fusion of logic "and" inhibits false

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press

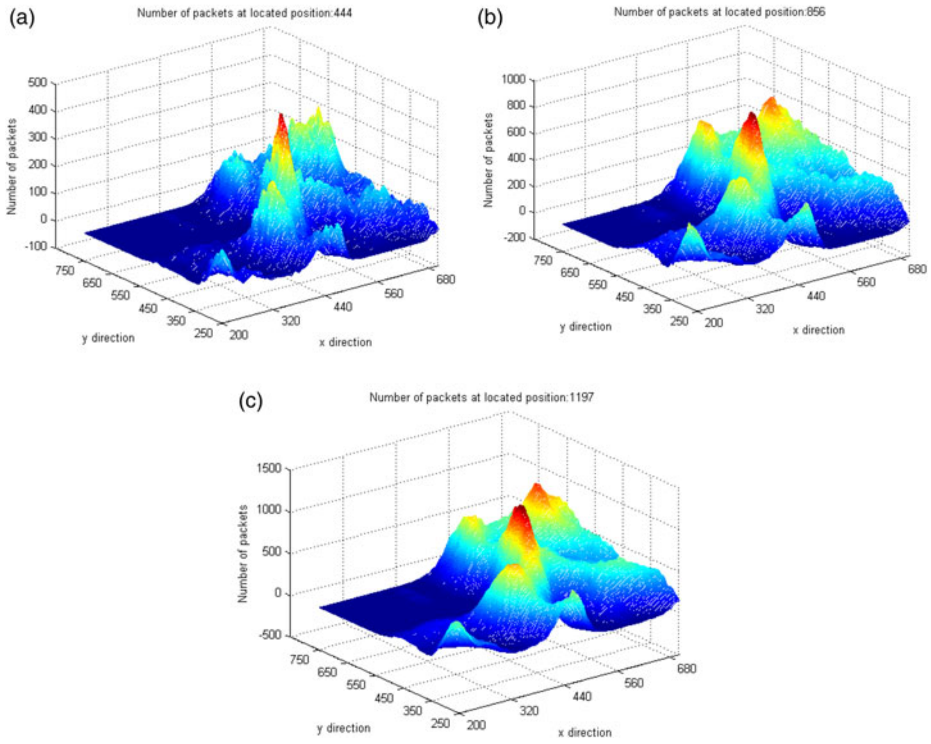


Figure 16. Distribution of the sum of SNDPs in the search area using NMIF, height value only, and surface features only as matching information. (a) NMIF (vector similarity fusion). (b) Height value only. (c) Surface features only.

peaks around the located point and the number of added packets is more focused. The advantage of NMIF is clear.

4.2.2. *Fusion based on vector similarity.* Vector similarity fusion, that is, thinking of the SNDP as a vector, compares two node similarities as two vectors (Sun, 2002; Zhang et al., 2009). Measuring the similarity of two vectors is usually equal to comparing their angle or length. First, we transform the two characteristic parameters of the node σ_1 and σ_2 into a two dimensional vector $\Gamma = (\sigma_1, \sigma_2)$. We use the Euclidean distance as the similarity measurement of two vectors, expressed as follows.

$$\Delta = \left((\Gamma_s - \Gamma_a)^T c^{-1} (\Gamma_s - \Gamma_a) \right)^{1/2} \tag{18}$$

where Γ_s and Γ_a are two-dimensional vectors composed of the node characteristic parameters, and c is a two-dimensional diagonal matrix representing the measurement noise of σ_1 and σ_2, the two elements of Γ_s and Γ_a. Therefore, the similarity is measured as follows:

$$\lambda_{ij} = \exp\left[-\frac{\Delta^2}{c^2} \right] \tag{19}$$

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press

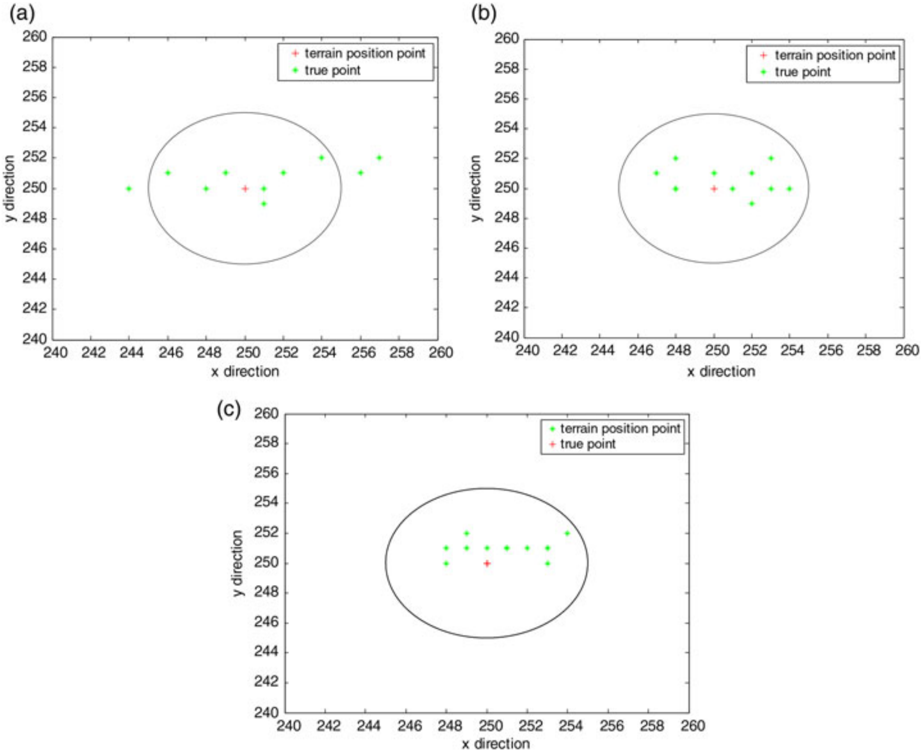


Figure 17. Positioning result distribution using three methods over 10 trials. (a) Height value only. (b) NMIF (vector similarity fusion). (c) NMIF (logic fusion).

$$\begin{cases} A_{ij} = \sum_{i=1}^{m} \sum_{j=1}^{n} \delta_{ij} \\ \delta_{ij} = \begin{cases} 1 & \lambda_{ij}^{\Delta} \geq \lambda_{threshold} \\ 0 & ealse \end{cases} \end{cases} \tag{20}$$

The experimental results for the DEM on a 6 m grid are shown in Figure 16. Figure 16(a) shows the distribution of the sum of SNDPs using vector similarity fusion, Figure 16(b) is the result using height value only, and Figure 16(c) is the result using surface features only.

From the results, we can see that for the same grid precision and measurement noise, the vector similarity method inhibited the false peaks around the located point better than the result using height value or surface feature information alone. The sum of the added SNDPs at the position point is more focused.

5. ACCURACY ANALYSIS.

As can be seen from the above results, NMIF can make full use of the information of a single node and its surrounding surface. The nearby pseudo-peaks surrounding the positioning node have been inhibited effectively.

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press

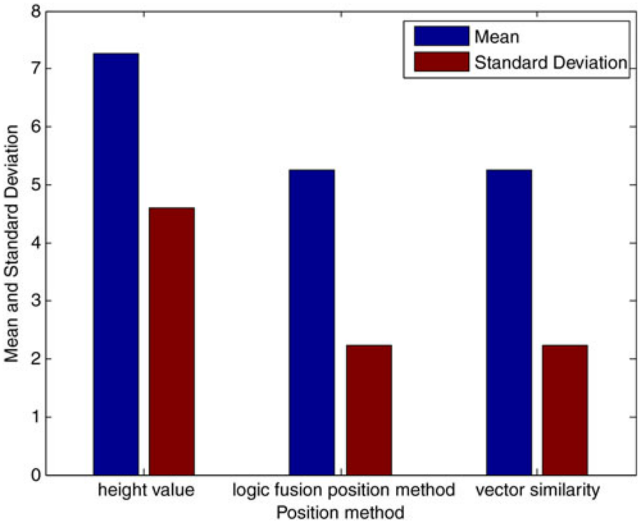


Figure 18. Mean value and standard deviation of the position deviation of the three methods.

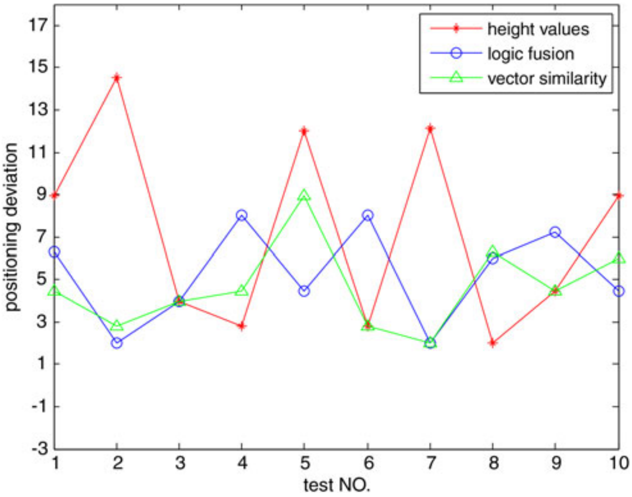


Figure 19. Position deviation of the three methods over 10 trials.

These results show that for the same terrain, NMIF obtains a more concentrated distribution of the number of packets, which is advantageous for positioning point identification, and inhibits pseudo-peaks. We now focus on analysing the precision and stability before and after using data fusion for positioning. In order to compare the precision and stability of the proposed method and the three comparison methods (no fusion, fusion based on vector similarity, and fusion based on logic) in a DEM with low precision and high measurement noise, we chose a DEM with a grid of 6 m and interpolated it into a 2 m grid. In addition, we added Gaussian measurement noise with zero mean, standard variance $\sigma = 1$, and circular error probability $\sigma_r = 0.5$. We

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press

computed the deviation for every experiment using $\Delta d = 2 \times \sqrt{\Delta i^2 + \Delta j^2}$. The results of the simulation output are as follows.

Figures 17(a)–(c) are the results for "height value," "logic fusion," and "vector similarity" for a grid of 6 m interpolated into a 2 m grid. Figure 18 compares the positioning error of the three positioning methods, and Figure 19 shows the mean and standard deviation of the positioning error for the three positioning method tests.

From the simulation positioning results, we can see that the positioning point distribution based on the NMIF method, both for vector similarity fusion and logic fusion, are more concentrated around the true location, and the mean and standard deviation of the positioning deviation are far smaller than the positioning result using height value only. It is clear that the NMIF method has better accuracy and stability than the height-based matching position method. NMIF extracts more than one terrain feature and the additional positioning information from the terrain surface increases the accuracy of the matching position stability.

5. CONCLUSIONS.

The essence of terrain matching positioning is a process that positions terrain features in the DEM. Previous methods consider the measured data as a discrete point sequence and hence cannot take full advantage of the measured information. As a result, they obtain a poor position result because of the lower amount of measured data or terrain features. NMIF is a good way to extract the terrain features, and the test results have shown the great advantages of NMIF at terrain matching positioning. We hence obtain the following conclusions. First, the MTM and DEM are not only point-wise discrete sequences, but form three-dimensional surfaces that include a large amount of terrain feature information. There is a determinate position relationship between the nodes, so we can extract more positioning information. Second, the experiments show that NMIF can extract diverse characteristics of the topography, increasing the utilisation ratio of the measured terrain to improve the accuracy and stability of the terrain matching position. Finally, as the measurement noise and topography grid side lengths increase, the position deviation will increase, but NMIF performs better than the position result using height or surface features alone. That is to say, the NMIF is more robust to measurement error.

ACKNOWLEDGEMENT

This project was supported by the National Natural Science Foundation of China (51279221).

REFERENCES

Anonsen, K.B. and Hagen, O.K. (2010). An analysis of real-time terrain aided navigation results from a HUGIN AUV. *OCEANS 2010*, Seattle, WA.

Carreno, S., Wilson, P., Ridao, P. and Petillot, Y. (2010). A survey on Terrain Based Navigation for AUVs. *OCEANS*, Seattle, WA.

Chen, X. (2013). *A Study on Underwater Terrain Matching Aided Navigation Technology of AUV*. Doctoral Thesis in Design and Construction of Naval Architecture and Ocean Structure, Harbin Engineering University, Harbin, China (in Chinese).

Hagen, O.K. (2006). TerrLab–a generic simulation and post-processing tool for terrain referenced navigation. *OCEANS*, Boston, MA.

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press

Hagen, O.K., Ånonsen, K.B. and Mandt, M. (2010). The HUGIN Real-Time Terrain Navigation System. *OCEANS 2010*, Seattle, WA.

Liu, C. (2003). *Underwater Vehicle Terrain Reference Positioning Technology Research*. Doctoral Thesis in Navigation, Guidance and Control, Harbin Engineering University, Harbin, China.

Meduna, D.K., Rock, S.M. and McEwen, R. (2010). *Closed-loop terrain relative navigation for AUVs with non-inertial grade navigation sensors.* Autonomous Underwater Vehicles, IEEE/OES, Monterey, CA. IEEE.

Meduna, D.K., Rock, S.M. and McEwen, R. (2008). *Low-cost terrain relative navigation for long-range AUVs.* Oceans, Quebec City, QC. IEEE.

Nakatani, T., Ura, T., Sakamaki, T. and Kojima, J. (2009). Terrain based localization for pinpoint observation of deep seafloors. Oceans, Bremen, Germany.

Nygren, I. (2005). *Terrain Navigation for Underwater Vehicles. Doctoral Thesis in Signal Processing*, Stockholm, Sweden.

Nygren, I. (2008). *Robust and Efficient Terrain Navigation of Underwater Vehicles. Position, Location and Navigation Symposium, IEEE/ION*, **1–3**, 923–932.

Shi, Y. (2003). *Probability theory and mathematical statistics.* Harbin: Department of Applied Mathematics, Harbin Engineering University (in Chinese).

Sun, X. (2002). *Modern Pattern Recognition*. Changsha: National Defense University Press (in Chinese).

Tian, F. (2007). *Research on Prior Map Data Processing Based Terrain Aided Navigation Methods for Underwater vehicle.* Master Degree Thesis of Engineering, Harbin Engineering University, Harbin, China (in Chinese).

Xie, Y.G. (2005). *Terrain aided navigation*. Master of Science Thesis, Stockholm, Sweden.

Zhang, F. Si, W., Yan, L. and Ge, Y. (2003). *Simulation Research on Matching Algorithm of Autonomous Underwater Navigation System*. Geomatics and Information Science of Wuhan University, 4:**28**(2) (in Chinese).

Zhang, H., Zhao, J. and Shao, N. (2011). *Underwater Navigation Method based on Characteristics of The Seabed Topography. Control Technology*, **30**(11), 96–102 (in Chinese).

Zhang, K. Zhao, J. and Chen, Z. (2014). *Research on Underwater Terrain Matching Navigation Based on Correlation. Journal Of Geodesy And Geodynamics*, 2:**34**(1) (in Chinese).

Zhang, W. and Yang, X. (2004). *Fuzzy Control Theory and Application*. Northwestern Polytechnical University (in Chinese).

Zhang, Y., Liu, Y. and Ji, Z. (2009). Vector similarity measurement method. *Technical Acoustics*, 8: **28**(4) (in Chinese).

https://doi.org/10.1017/S0373463316000369 Published online by Cambridge University Press