Application of a Graphical Image Pre-Retrieval Method Based on Compatible Rough Sets to the Self-localization Method of Mobile Robots

Chaohua Yan
School of Digital Media and Art Design Nanyang Institute of Technology Nanyang, Henan, China,473004
E-mail: xingtian@nyist.edu.cn

Keywords: compatible rough set, image pre-retrieval, image processing, mobile robot, autonomous positioning

Recieved: December 2, 2023

With the gradual maturation of mobile robot technology, mobile robots are playing an important role in more and more fields, also from the former industrial supplies to thousands of households, and the premise of mobile robot's application in these fields is that mobile robots can achieve positioning autonomously. Therefore, it is of great importance to study autonomous localisation techniques for mobile robots. In this research, we develop a visual localization system for mobile robots and explore the application of a graphical picture pre-retrieval method based on compatible rough sets to the self-localization of mobile robots. The key function of the system is autonomous localisation, based on a known map, and global localisation needs to be completed before specifying a target point. To address the shortcomings of traditional global localisation which requires human intervention to complete, global localisation without human intervention is achieved by combining a UWB module and two improved methods, in the navigation process, based on the localisation information obtained by the EKF algorithm, the adaptive Monte Carlo algorithm fuses In the navigation process, based on the positioning information obtained by the EKF algorithm, the positioning accuracy is further improved by the adaptive Monte Carlo algorithm fusing the odometry, IMU, UWB and LIDAR to meet the requirements for high positioning accuracy in complex environments.

Povzetek: Razvita metoda za iskanje slik na osnovi združljivih grobih množic izboljša samolokalizacijo mobilnih robotov z visoko natančnostjo in hitrostjo zaznavanja v kompleksnih okoljih.

1 Introduction

Mobile robot positioning technology has a history of several decades in foreign countries, but the scope and depth of its application varies due to different national conditions, and is currently focused on industrial and agricultural fields [1]. The typical representative of mobile robot localization technology is based on image matching. The method is based on the fusion, retrieval and classification of target images to achieve autonomous positioning of the robot, using visual tentative estimation to analyse, predict or describe the interrelationship between objects [2], and to build a position-time model of the robot to optimise the positioning process. Self-positioning techniques enable mobile robots to obtain position information both indoors and outdoors and transmit this data to the controller, enabling the robot to obtain accurate information about its surroundings [3]. Images provide a wealth of information and therefore the amount of data in an image is relatively large [4]. In practice, computing every pixel of an image consumes a very large amount of computational performance, and since many of the pixel points are meaningless, computing every pixel also wastes a lot of computational effort on many pixels that do not carry much information [5]. It is clearly more sensible to process not all the pixels in an image, but rather some points in the image that carry a lot of information. The extraction of these information-rich points is therefore a very important issue [6]. Image retrieval technology is a new research tool that combines computer vision, computation and pattern recognition methods. In practice, it can be used to extract information from images and to locate targets using graphical features [7-8]. Table 1 explains the summary of relevant studies.

Table 1: Summary of relevant studies

References	Method	Objectives	Summary of findings
[9]	Content-Based Remote Sensing Image Retrieval (CBRSIR)	The study suggested feature and hash (FAH) learning, a novel approach for CBRSIR that combines adversarial hash learning (AHLM) with deep feature learning model (DFLM).	To ensure retrieval precision, the DFLM learns the dense characteristics of the RS pictures.
[10]	Fuzzy sets	The study employed a variety of statistical modeling techniques, including data granulation and fuzzy sets, picture keypoint identification and comparison, data categorization and grouping, and simulation modeling.	The study applied relevance is supported by the data processing time criterion without compromising its dependability and interference immunity qualities.
[11]	Bag-of-words	The study suggested an enhanced bag-of-words form for the characterisation of activity signals.	Discrete wavelet transformation is used to process the raw activity signals in order to extract local features. These features are then clustered using K-means in order to create a dictionary of bag-of-words.
[12]	Convolution neural network (CNN)	The study carried out hierarchical localization of a mobile robot in an indoor setting. The CNN receives unidirectional images as input.	The original CNN is converted into a regression CNN using a transfer learning technique, which enables it to predict the coordinates of the robot's position within a given room.
[13]	Drift diffusion model (DDM)	The study determines whether retrieval and non-decision process speeds were correlated with retrospective confidence in recognition tests.	The study examined data from six previous trials. It discovered that, in the DDM, higher retrospective confidence ratings were linked to both larger drift rates and shorter non-decision times.
[14]	Nonmonotone neural network	The Study suggested a straightforward self-localization technique. According to the suggested approach, a nonmonotone neural network's state space contains the order structure, such as the mobile robot's navigation path, embedded as trajectory attractors.	Processing based on the network's autonomous dynamics is then used to estimate the mobile robot's position.
[15]	Adaptive Monte Carlo Localization (AMCL)	The research integrates AMCL with the Split-and-Merge line detection technique. To find localization errors, the identified line segments are compared with the surrounding walls.	A portion of the particles is placed at the error's position to modify the robot's attitude when one is identified. The robot can re-locate itself in this manner.

2 Graphical image retrieval methods based on compatible rough sets

2.1 Compatible rough set model

Some scholars have proposed a generalized model based on rough set theory that substitutes the compatibility relation for the equivalency relation. In other words, the equivalency relation is weakened to become the compatibility relation, which only satisfies the self-reflexivity and symmetry. The research process will then reveal whether there is a relationship between the interrelated attributes. When judgments are made in an information system, it is discovered that there is a tiny variation in the attribute values, and that the variation is insignificant, which enhances fault tolerance.

Definition 1: In an incomplete information system, with the following equation1 and possible null values in A, and noted as "*", the tolerance relation T(A) is defined as equation 2.

$$IS = \langle U, V, AT, F \rangle, AT = C \cup D, \quad A \in AT \tag{1}$$

$$T(A) = \{(x, y) \in U^2 : \forall a \in A, a(x) = a(y) \vee a(x) = * \vee a(y) = *\} \tag{2}$$

Definition 2: In the theoretical domain U, let T be a binary relation on a finite, non-empty domain. A relation T is said to be compatible if it satisfies self-reflexivity and symmetry, i.e. the following two conditions

$$\forall x \in U, xTx; \forall x, y \in U, xTy, yTx \tag{3}$$

Definition 3: Let the space of compatible approximations (U, T), Let T be a compatible relation on U and U be a finite, non-empty theoretical domain. For any X belonging to U, the upper and lower approximations on the compatible relation T are defined by equations (4) and (5), and then equation (6) is called the compatible rough set model.

$$\overline{apr}_\tau(X) = \{x \in U | [x]_\tau \cap X \neq \emptyset\} \tag{4}$$

$$\underline{apr}_\tau(X) = \{x \in U | [x]_\tau \subseteq X\} \tag{5}$$

$$\left(\underline{apr}_T(X), \overline{apr}_T(X)\right) \tag{6}$$

From the definition, if the compatibility relation satisfies the transfer relation, the compatibility relation becomes an equivalence relation and the compatibility rough set model becomes a rough set model. The compatibility approximation space (U, T) can be divided into mutually disjoint positive, negative, and boundary domains, respectively, using the upper and lower approximations of the compatibility rough set.

$$pos_T(X) = apr_T(X) \tag{7}$$

$$neg_\tau(X) = U - \overline{apr}_\tau(X) \tag{8}$$

$$bnd_\tau(X) = \overline{apr}_\tau(X) - \underline{apr}_\tau(X) \tag{9}$$

Then, the upper and lower approximations of compatible rough sets satisfy the following requirements.

$$\underline{apr}_T(X) \subseteq \overline{apr}_T(X);$$

$$\underline{apr}_T(\mathcal{O}) = \overline{apr}_T(\mathcal{O}) = \emptyset;$$

$$\underline{apr}_T(U) = \overline{apr}_T(U) = U;$$

$$\underline{apr}_\tau(X \cap Y) \subseteq \underline{apr}_\tau(X) \cap \underline{apr}_\tau(Y);$$

$$\overline{apr}_\tau(X \cup Y) \supseteq \overline{apr}_\tau(X) \cup \overline{apr}_\tau(Y);$$

$$\underline{apr}_\tau(X \cup Y) \supseteq \underline{apr}_\tau(X) \cup \underline{apr}_\tau(Y);$$

$$\overline{apr}_\tau(X \cap Y) \subseteq \overline{apr}_\tau(X) \cap \overline{apr}_\tau(Y).$$

Definition 4: Let U be a theoretical domain, the compatibility relation on U be T, and any X belongs to U. The following notation represents the set X's approximate accuracy and roughness with regard to the compatibility relation T.

$$\eta(X) = \frac{|apr_T(X)|}{|apr_T(X)|} \tag{10}$$

$$\omega(X) = 1 - \eta(X) = 1 - \frac{|apr_T(X)|}{|apr_T(X)|} \tag{11}$$

Findings indicate that the rough set model, which is based on the compatibility relation study, can enhance decision accuracy and offer a fresh approach to the study of incomplete information systems for mobile robots. Pseudocode 1 for Graphical Image Retrieval Method Based on Compatible Rough Sets (GIRC) is as follows.

Pseudocode 1: Graphical image retrieval method based on compatible rough sets (GIRC)

```
Function GIRC (image_database, query_image):
    query_features = extract_features(query_image)
        database_features = []
    for each image in image_database:
        database_features. append(extract_features(image))
        lower_approximation =
compute_lower_approximation (query_features,
database_features)
    upper_approximation = compute_upper_approximation
(query_features, database_features)
    similarity_scores = []
    for each image_features in database_features:
        similarity_scores. append (calculate_similarity
(query_features, image_features))
        sorted_indices =
sort_indices_by_similarity(similarity_scores)
    retrieved_images = []
    for each index in sorted_indices:
        retrieved_images. append(image_database[index])
    return retrieved_images
function extract_features(image):
```

```
    return features
function compute_lower_approximation (query_features,
database_features):
    return lower_approximation_set
function compute_upper_approximation (query_features,
database_features):
    return upper_approximation_set
function calculate_similarity (query_features,
image_features):
    return similarity_score
function sort_indices_by_similarity(similarity_scores):
return sorted_indices
```

2.2 Compatibility rough set theory for grey-scale image pre-retrieval

The definition of an image's rough entropy is as follows: as Figure 1 below illustrates, different objects and backgrounds have distinct distributions of REt values for different roughness values, and Figure 2 shows the points of REt on the plane.

$$RE_T = -\frac{e}{2}\left[R_{O_T}\log_e\left(R_{O_T}\right) + R_{B_T}log_e\left(R_{B_T}\right)\right] \quad (12)$$

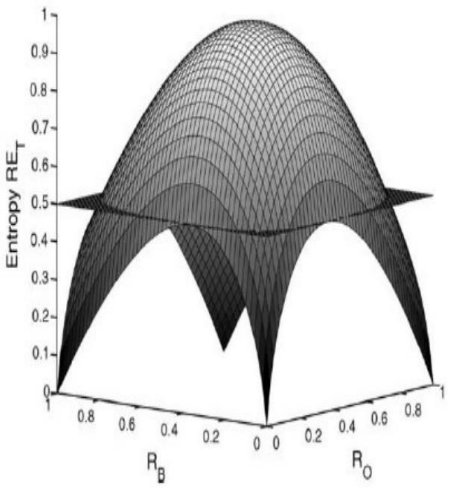

Figure 1: Distribution of REt values for different roughness values

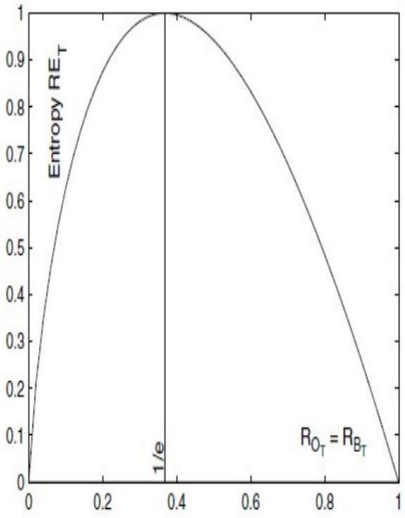

Figure 2: Points of REt on the plane

Image objects can be extracted by minimising the roughness of the object and background areas. For each grey level T that represents the background and object areas of the image, a REt value is calculated and the one that makes the RET the largest is chosen, in other words, the optimal threshold for object-background separation is determined to be T *. The following is a description of the algorithm for selecting T *.

Initialization: Four integer arrays $object\_lower = O\_l;\ object\_upper = O\_u, background\_upper = b\_u\ are\ 0, Background\_lower = b\_lower.$

$Step\ 1: for\ i = 1\ to\ number\_g$

max_gi
$= the\ maximum\ grey\ value\ of\ the\ pixel\ points\ in\ gi$

min_gi
$=\ the\ minimum\ grey\ value\ of\ the\ pixel\ in\ gi$

$(a)\ For\ max\_gi \le j \le max\_gray$

$O\_l(j) = O\_l(j) + 1$

$(b)\ For\ min\_gi \le j \le max\_gray$

$O\_u(j) = O\_u(j) + 1$

$(c)\ For\ min\_gray \le j \le min\_gi$

$$B\_l(j) = background(j) + 1$$

$$(d)\ For\ min\_gray \le j \le max\_gi$$

$$B\_u(j) = b\_u(j) + 1$$

$$Step\ 2: for\ l = min\_gray\ to\ max\_gray$$

$$Object\_roughness\ (l) = 1 - [o\_l(l)/O\_u(l)]$$

$$Background\_roughness\ (l) = 1 - [b\_l(l) / b\_u(l)]\ Rough\_entropy(l)$$

$$Step\ 3: Threshold\ (optimal) argmax$$

$$[rough\_entropy(l)]$$

Note that given the values of max_ gray and min_ gray, the computation of the rough entropy requires only one scan of the image pixel points, because for any value of i, max_gannulei and min_granulei are only computed once.

To give a target image that is similar to the query image, the similarity between the query and target images is computed. The similarity value, which ranges from 0 to 1, is then used to calculate the degree of similarity between the target image and the query image. Larger values are considered to be similar to the query image, and when the value of similarity is 1, the target image is considered to be identical to the query image. Firstly, the user inputs the query image and determines an optimal threshold value to separate the image object from the background. In the process, two integer arrays concerning the gray values are really obtained, together with the set of upper and lower approximations representing the objects. Therefore, in order to match objects in two images, it is necessary to compare the upper and lower approximation arrays of the two image objects. Here, the author defines the OSR based on the similarity between image objects as follows:

$$\mathrm{OSR} = \mathrm{N}/\mathrm{T} \quad (13)$$

Where N is the number of pixels with the same value in the upper and lower approximation sets of the image object and T is the size of the lower approximation set of the object. It can be found that the OSR takes values between [0, 1], and when the OSR is close to 1, it means that the two images are identical that the two images are identical. Compared to those classical methods of image similarity calculation, OSR avoids the high dimensional space of image feature vectors and thus does not suffer from dimensional catastrophe. The complexity is reduced and, with it, there are substantial savings in computing time and storage costs. In its form, OSR is based on the definition of statistical properties as a ratio relation and is therefore more robust in terms of image rotation, dimensionality and transformation.

Rough set theory can be used to mine knowledge from the data in order to build pattern recognition models for images. A framework for image recognition and retrieval based on rough set theory is shown in Figure 3.

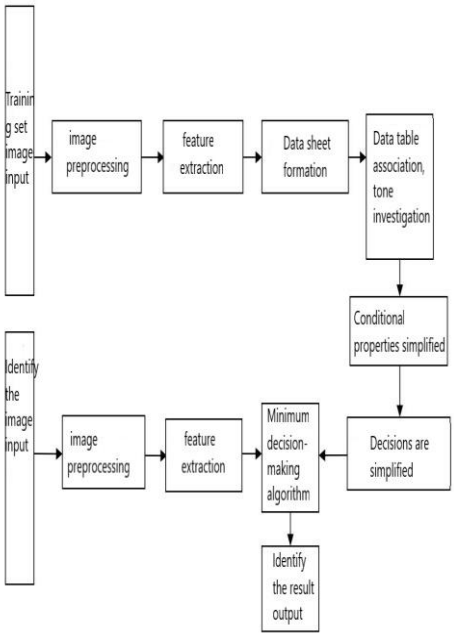

Figure 3: Recognizing and retrieving images using crude set theory

Statistical analysis

Data gathering, organizing, analyzing, interpreting, and presenting are the phases in statistical analysis. Tests for logistic regression are used.

Logistic Regression analysis

It uses probabilities related to Y values to describe the response variable. (y) Denotes the likelihood that Y = 1, which is ROP. Similarly, 1-(x) is the likelihood that Y= 0, which denotes the absence of ROP, would occur. The following is how these probabilities are expressed:

$$\pi(x) = D(Y = 1|X_1, X_2, .., X_m) \quad (14)$$

$$1 - \pi(x) = D(Y = 1|X_1, X_2, .., X_n) \quad (15)$$

$$Jm\frac{D(Y=1|X_1,X_2,..,X_n)}{1-D(Y=1|X_1,X_2,..,X_n)} = Jm\frac{\pi(x)}{1-\pi(x)} = \beta_0 + \sum_{j=1}^{n}\beta_j j \quad (16)$$

Using the log it transformation of equation (3)'s inverse, we obtain:

$$D(Y = 1|X_1, X_2, .., X_n) = \frac{l^{\beta_0+\sum_{j=1}^{n} j}}{1+l^{\beta_0+\sum_{j=1}^{n}\beta_j X_j}} = \frac{1}{1+l^{-(\beta_0+\sum_{j=1}^{n}\beta_j X_j)}} \quad (17)$$

They used logistic regression to fit the data. We first require a strategy for forecasting the parameters. Parameter estimation uses maximum likelihood.

$$\zeta(X_i) = \pi(X_i)^{y_i}[1 - \pi(X_i)^{1-Y_i}] \quad (18)$$

An example of data is all that this equation takes into account. To obtain the whole likelihood function, we may multiply the likelihood contributions of the observations assuming that they are independent. Equation (19) results in the following:

$$R(E) = \prod_{i=1}^{n}\zeta(X_i) \quad (19)$$

Utilizing maximizing $R(E)$, one may obtain maximum likelihood estimates. The logarithm is used to maximize the likelihood function. Log-likelihood is denoted by the symbol $r(E)$ in equation (8).

$$V = -2Im[r(E)] \quad (20)$$

The issue can be resolved by computer software. The logistic regression analysis will use SAS to get the maximum likelihood estimates for this project.

$$\sum_{i=1}^{n} X_{if}[Y_i - \pi(X_i)] = 0, \quad forF = 1,2 \ldots\ldots, j \quad (21)$$

The remainder of this study examines SAS's calculated logistic regression model parameters. The logistic function sometimes called the sigmoid function is how the equation is expressed. It may be stated as follows:

$$b = 1 / (1 + a\hat{}(-h)) \quad (22)$$

Where b=is the dependent variable's positive class probability

a=is natural logarithm's base

h=is the weighted linear combination of independent variables

3 Test results and analysis

In our daily life and work, people have high demands on image quality and therefore need to improve its processing. From image acquisition and transmission to storage and use, image pre-processing is indispensable. For a complete and reliable information, only the parts that do not meet the criteria can be called valid information if they are properly filtered out. Those that cannot be used directly or are of low usability are partially removed to ensure that the quality of the acquired images meets the requirements. In addition, the image information is often interfered with by various noise sources during the acquisition process, which can also have an impact on the positioning of the mobile robot if not removed.

3.1 Image filtering

Tone method that has been used in the field of image processing is denoising, which is known as filtering. The quality of an image is determined by a number of factors, one of which is the filtering algorithm, of which the filter is the most important. A filter is an operation that classifies and encodes images with different properties and characteristics. This can be achieved by removing noise or text information from the images, or by using it to remove certain frequency components with a higher probability density for certain purposes (e.g. image segmentation, blurring), thus improving the signal-to-noise ratio. In general, in order to improve the image processing system to achieve the target value for accurate and rapid detection, tracking and identification, are required to use some of the better filters to ensure quality; the second is that there are many ways of smoothing, which is the most effective and most used is the grayscale transformation to a state of no bias [16, 17].

3.2 median filtering

Median filtering is a digital image for the elimination of noise and interference, and through the design of one or more filters to suppress or reduce the image signal in certain frequency bands on the effective method of "noise". It is generally composed of a standard frequency source (such as a smoothing function), high frequency sampling in the frequency domain and odd-order spatial analysis. The low-pass filter is very sensitive to noisy speech because of its wide bandwidth. However, median filtering can only remove a portion of the noise and interference factors. The mathematical expressions of one- and two-dimensional median filtering are as follows.

$$y_i = Med\{f_{i-v}, \cdots, f_i, \cdots, f_{i+v}\}, i \in Z, v = \frac{m-1}{2} \quad (23)$$

$$y_{ij} = Medf_{ij} \quad (24)$$

The shape and size of the window of the 2D median filter has a large impact on the filtering effect, and different window shapes and sizes are often used for different image contents and different application requirements. The robot image in this paper requires ensuring that the marker edges are clear and that the details of the marker such as points, lines and cusps are retained as much as possible, so median filtering is not appropriate.

3.3 Gaussian smoothing

An ideal smoothing filter with excellent performance is one that uses a Gaussian kernel. The smoothing filter's shape has an impact on how well it performs. If the spatial domain error is set to Δx, the mistake in the frequency domain as Δω. They are related as shown in equation (16).

$$\Delta x \Delta \omega \frac{1}{4\pi} \quad (25)$$

The choice of the optimal filter is to optimise this relationship. Filtering with a Gaussian filter can be achieved by convolution in the time domain (or spatial domain), or by multiplication in the frequency domain. Its one-dimensional distribution function, two-dimensional distribution function, and two-dimensional distribution function correspond to discrete forms as

$$G(x) = \frac{1}{2\pi\sigma} exp\left(-\frac{x^2}{2\sigma^2}\right) \quad (26)$$

$$G(x, y) = \frac{1}{2\pi\sigma^2} exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \quad (27)$$

$$W(i, j) = \frac{1}{2\pi\sigma^2} exp\left(-\frac{i^2+j^2}{2\sigma^2}\right) \quad (28)$$

If the original image is represented by I and the filtered image is represented by I ', then

$$I' = I * W \quad (29)$$

This paper filters the scene image using a Gaussian smoothing filter. This is easy to use and quick, and it avoids complicated mathematical computations. The results of the processing are shown in Figure 4 below.

Figure 4: Gaussian smoothing effect (left image after processing)

4 Image segmentation - grey-scale threshold segmentation method

The grey-scale threshold segmentation method is one of the basic methods in the field of image processing, and it has many advantages in practical applications. Although the grey value segmentation method can directly obtain the target area after pre-processing operations such as smoothing and binarisation of the image [18]. However, it is difficult to apply it to large amounts of data due to the complexity of the threshold selection, the multi-level nature and the problem of fusion of information from multiple sources. The main principle of grey-scale thresholding is to treat each pixel in an image as a unified whole, and then process each region according to its grey-scale value. Assuming that t is a threshold and all points less than t are called target object points, while all points greater than or equal to t are called background points, the probability of incorrectly segmenting a background point as an object point is

$$E_1(t) = \int_{-\infty}^{t} q(l)dl \quad (30)$$

and the probability of mis-segmentation where the object points are misclassified as background points is

$$E_2(t) = \int_t^{\infty} p(l)dl = 1 - \int_{-\infty}^{t} p(l)dl \quad (31)$$

The total mis-segmentation probability is

$$E(t) = PE_2(t) + QE_1(t) = P\left(1 - \int_{-\infty}^{t} p(l)dl\right) + QE_1(t) \quad (32)$$

In order to find the threshold t when the classification error is minimum, the corresponding formula can be derived and the derivative can be made to be 0. After taking the logarithm, the value of t can be found as follows

$$p(t) = \frac{1}{\sqrt{2\pi}\sigma_1}\left[\frac{-(t-\mu_1)^2}{2\sigma_1^2}\right] \quad (33)$$

$$q(t) = \frac{1}{\sqrt{2\pi}\sigma_2}\left[\frac{-(t-\mu_2)^2}{2\sigma_2^2}\right] \quad (34)$$

$$\ln\sigma_1 + \ln Q - \frac{(t-\mu_2)^2}{2\sigma_2^2} = \ln\sigma_2 + \ln P - \frac{(t-\mu_1)^2}{2\sigma_1^2} \quad (35)$$

$$\sigma_2^2(t-\mu_1)^2 - \sigma_1^2(t-\mu_2)^2 = 2\sigma_1^2\sigma_2^2 \ln\frac{\sigma_2 P}{\sigma_1(1-P)} \quad (36)$$

When in a uniformly illuminated light environment, the above method can be used to find the optimal threshold.

Vision positioning system for mobile robots based on image retrieval.

General architecture of the system: In order to test the feasibility of the system, a mobilerobot platform needs to be built. The built robot platform is equipped with a Nano computer platform with Ubuntu 18.04, and the algorithm implementation of the whole system is done by this computing platform. The LIDAR RPLIDAR A1, the Astra Pro camera and the UWB positioning module all communicate with the computing unit via a USB 2.0 interface, as shown in Figure 5.

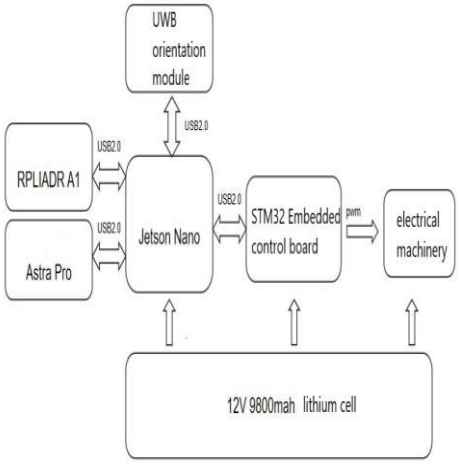

Figure 5: Hardware architecture

The framework of the system design is shown in Figure 6 below, with the main research elements being the SLAM algorithm module, its own localisation module and the path planning module, combined with the relevant sensors.

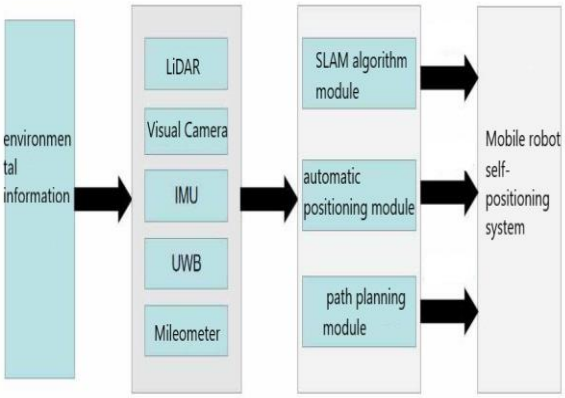

Figure 6: System design diagram

Global autonomous positioning scheme design: To address the problem of global positioning requiring human intervention, this paper uses the data from the UWB module to provide the initial position information and fuses it with the improved adaptive Monte Carlo algorithm to achieve global positioning of the mobile robot without human intervention.

The adaptive Monte Carlo algorithm is divided into two main improvements: the creation of a relevant document that records the real-time position of the vehicle and, instead of defaulting to a starting position of 0 at each navigation start, the documented position is recorded. Enhancing the adaptive Monte Carlo algorithm's particle update mechanism to allow real-time updating based on LIDAR observations even while the robot is not moving, rather than requiring global localization to be started only when it is moving. Both of these improvements allow for autonomous global positioning of mobile robots in indoor environments in some specific situations. However, in some other cases, the localisation fails. For example, in the first improvement, real-time position changes are only recorded when the mobile robot starts navigation, and the positioning information is inaccurate if the robot is moved artificially when navigation is not on or when the power is off. The second improvement has a certain probability of failure for autonomous positioning in some large scenarios or when the position on the map is far from the real position. This paper therefore proposes a way to integrate the UWB module with these two improvements to achieve autonomous global positioning without human intervention [19]. Robotics hardware, such as LIDAR and UWB modules, gathers environmental data. By generating a 3D image of the surroundings, LIDAR sensors shoot laser beams and time how long it takes for the beams to reflect off things. On the other hand, UWB modules provide exact localization by measuring the distance between the robot and other UWB-equipped objects via radio waves. Software algorithms use sensor fusion techniques to combine data from various sensors, such as LIDAR and UWB. In order to get a more complete picture of the robot's surroundings and its own location within them, sensor fusion integrates data from several sensors. Robot location and orientation in relation to environment are determined by software algorithms using information from LIDAR and UWB modules. Real-time pose (position and orientation) estimation of robots is achieved by the processing of sensor data by localization algorithms including Monte Carlo Localization (MCL), SLAM (Simultaneous Localization and Mapping), and Extended Kalman Filters (EKF).

The system inputs the UWB module and the documentation module, and by comparing the two positions, if the difference between the two is more than 10cm, the indoor robot is judged to have been moved by humans without navigation, believing the position data from the UWB module, and then automatically updating the particles to determine the attitude information. Algorithm can easily be updated to the correct attitude. If the difference between the two distances does not exceed 10cm, then the pose data recorded in the documentation is trusted, in which case the pose of the robot in the map does not move and the documented pose data is used directly.

If the robot is moved by a small amount, the automatic update particle scheme will update back to the correct pose more quickly. By using the above steps, the robot is able to autonomously and quickly obtain an accurate position in an indoor environment when the robot is switched on, regardless of how far away the cart is from the starting point of the map.

4.2 Documenting the navigation posture

The author uses a document to record the position of the mobile robot each time the navigation is switched off. The documented position is then used as the starting position for the mobile robot, eliminating the need to move the mobile robot back to the starting point or to manually adjust the position of the cart. This document is used to record the real-time attitude of the mobile robot, which is obtained by listening to a real-time conversion from the base_link tf coordinates to the map tf coordinates, and finally reading the attitude of the mobile robot on the map and writing it to the text file we have created in the path. As the robot is moving in a plane, it does not need to consider the translations on the Z-axis in the Cartesian coordinate system, but only the translations on the X- and Y-axes, and the angles are simply taken from the quaternions z and w, with w being the coefficient in front of the real part of the quaternion and z being the coefficient in front of the imaginary part of the quaternion, where the two quaternions can be treated as complex numbers in a two-dimensional plane. For the initial pose, i.e. the starting point of the construction, the values are simply set to x=0, y=0, z=0 and w=1.

The function of listening to the tf is mainly implemented by a new function package, which creates a publisher that listens to the relative relationship between the base_link coordinate system and the map coordinate system, i.e. the position of the base_link coordinate system under the map coordinate system, and then writes the pose information to the document created earlier. Once the pose information has been written to the document, it needs to be called each time navigation is started. The document's pose information is written to the AMCL function pack as the initial pose for navigation. By writing this node in the launch file when the navigation is started, the position information of the mobile robot with respect to the map is automatically stored in the

document when the navigation is run, and by writing a procedure to read the coordinates in the Request Map function in the acml_node.cpp file in the AMCL function package when the navigation is started a second time, the position information in the document is automatically used for the second navigation. The position information in the document is automatically used as the initial position for the second navigation. The program was recompiled and the navigation launch test was opened, with the mappose node written into the AMCL function package. The test was successful as follows, and the mappose topic can be called in the rviz visualisation software to record the mobile robot's real-time poses.

4.3 Automatic particle update improvement

In the original program, the decision to update a particle is made in the Laser Received function based on whether the particle's pose has changed or not. In this paper, an or condition is added to the judgment condition to update the particle, so that the particle will be updated if only one of the conditions is met. If the covariance of the robot's position is less than 0.001 is the next step if the velocity on the x, y, and z axes is 0. Some of the code improvements are as follows.

```
updatePoseFromServer ();
geometry_msgs: PoseWithCovarianceStamped pose;
pose = readpoint ();
pf_vector_t pf_init_pose_mean = pf_vector_zero ();
pf_init_pose_mean. v [0] = pose. pose. position.x;
pf_init_pose_mean. v [1] = pose. pose. pose. position. y;
pf_init_pose_mean. v [2] = tf2: getYaw (pose. pose. pose. orientation);
pf_matrix_t pf_init_pose_cov = pf_matrix_zero ();
pf_init_pose_cov.m [0][0] = 0.5 * 0.5;
pf_init_pose_cov.m [1][1] = 0.5 * 0.5;
pf_init_pose_cov.m [2][2] = (M_PI/12.0) * (M_PI/12.0);
pf_init (pf_, pf_init_pose_mean, pf_init_pose_cov);
pf_init_ = false;
```

5 Discussion

CBRSIR [9] depends critically on the quality of features extracted from pictures. Retrieval performance might not be at its best if the features do not accurately reflect the content of the images. The fuzzy [10] rules and membership functions used to depict picture similarity are sometimes intricate and challenging to comprehend, fuzzy set approaches may be uninterpretable. The comprehension of image content is restricted by Bag of Words [11] approaches, which interpret images as collections of visual words and fail to capture the semantic links between image elements. To extract discriminative features from images, CNN [12] techniques frequently need a large amount of computational power and training data, which makes them less appropriate for situations with limited resources. To learn discriminative features for image retrieval, DDM algorithms require labeled training data, which can be expensive and time-consuming to obtain, particularly for large-scale datasets [13]. The interpretability of Non-Monotone Neural Networks [14] may be compromised by their intricate designs and high-dimensional representations. The precision and dependability of sensor data, such as those from odometers and laser range finders, are critical to AMCL [15] performance and can be impacted by hardware constraints as well as ambient conditions. The Graphical Image Pre-retrieval Method was suggested as a solution to these problems.

6 Conclusion

The development of human society is inseparable from the progress of technology, and as an intelligent machine, the information retrieval technology of mobile robots has significant advantages such as very high accuracy, fast and convenient. In terms of the positioning module of the system, this paper integrates odometry, IMU and UWB modules to improve the accuracy of positioning during map building, and after the completion of the map building, based on the known map, the trajectory tracking and global positioning are investigated. In terms of global positioning, initial position information is provided by the UWB positioning module, and then the attitude information is provided by both the document recording navigation position and the automatic update of particles to achieve autonomous global positioning without human intervention.

Data availability

All data are included within the article.

Conflicts of interest

The authors declare no conflicts of interest.

Funding statement

1.2024 International Science and Technology Cooperation Project of Henan Province, Project No. 242102520027, Project Name: Research and application of lung cancer CT Identification based on DCNN optimization;

2. Education and Teaching Reform Research Project of Nanyang Institute of Technology, Project No. NIT2023JY-061, Project Name: Exploration of the course teaching reform of C # Programming Design of Digital Media Technology major based on interdisciplinary integration

References

[1] Panigrahi PK, Bisoy SK. Localization strategies for autonomous mobile robots: A review. Journal of King Saud University-Computer and Information Sciences. 2022 Sep 1;34(8):6019-39.

[2] Zhou Zong Kun, Jiang Weiping, Tang Jian, et al. LiDAR map matching and 2D code fusion for indoor positioning and navigation of AGVs[J]. Mapping Bulletin, 2021(01): 9-12+52.

[3] Zhang, P., Li, T., Wang, G., Luo, C., Chen, H., Zhang, J., Wang, D. and Yu, Z., 2021. Multi-source information fusion based on rough set theory: A review. Information Fusion, 68, pp.85-117.

[4] Choudhuri, R., Halder, S. and Halder, A., 2021, November. A Novel Rough Set Based Image Denoising Algorithm. In 2021 Sixth International Conference on Image Information Processing (ICIIP) (Vol. 6, pp. 19-24). IEEE.

[5] Ismael, A.A. and Baykara, M., 2021. Digital Image Denoising Techniques Based on Multi-Resolution Wavelet Domain with Spatial Filters: A Review. Traitement du Signal, 38(3).

[6] Wang, W., Li, H., Wang, K., He, C. and Bai, M., 2020. Pavement crack detection on geodesic shadow removal with local oriented filter on LOF and improved Level set. Construction and Building Materials, 237, p.117750.

[7] Liu Honglin. Research on mobile robot localization based on improved particle filtering[D]. Anhui University of Engineering, 2020.

[8] Zhang, S., Tan, X. and Wu, Q., 2021, December. Self-positioning for mobile robot indoor navigation based on wheel odometry, inertia measurement unit and ultra wideband. In 2021 5th International Conference on Vision, Image and Signal Processing (ICVISP) (pp. 105-110). IEEE.

[9] Liu, C., Ma, J., Tang, X., Liu, F., Zhang, X. and Jiao, L., 2020. Deep hash learning for remote sensing image retrieval. *IEEE Transactions on Geoscience and Remote Sensing*, 59(4), pp.3420-3443.

[10] Daradkeh, Y.I., Tvoroshenko, I., Gorokhovatskyi, V., Latiff, L.A. and Ahmad, N., 2021. Development of effective methods for structural image recognition using the principles of data granulation and apparatus of fuzzy logic. *IEEE Access*, *9*, pp.13417-13428.

[11] Bhuiyan, R.A., Tarek, S. and Tian, H., 2021. Enhanced bag-of-words representation for human activity recognition using mobile sensor data. *Signal, Image and Video Processing*, 15(8), pp.1739-1746..

[12] Ballesta, M., Paya, L., Cebollada, S., Reinoso, O. and Murcia, F., 2021. A cnn regression approach to mobile robot localization using omnidirectional images. *Applied Sciences*, 11(16), p.7521.

[13] Hu, X., Yang, C. and Luo, L., 2022. Retrospective confidence rating about memory performance is affected by both retrieval fluency and non-decision time. *Metacognition and Learning*, 17(2), pp.651-681.

[14] Hu, W. and Zhu, Q., 2022. Spatial-temporal dynamics of a non-monotone reaction-diffusion Hopfield's neural network model with delays. *Neural Computing and Applications*, 34(13), pp.11199-11212.

[15] Bontan, L., 2023. Line Adaptive Monte Carlo Localization: Improving self-localization of a mobile robot in barns.

[16] Zhang, H., Wen, J., Liu, Y., Luo, W. and Xiong, N., 2020. Mobile Robot Localization Based on

Gradient Propagation Particle Filter Network. *IEEE Access*, 8, pp.188475-188487.

[17] Gao, X. and Zhang, T., 2021. Introduction to visual SLAM: from theory to practice. Springer Nature.

[18] Ravankar, A.A., Ravankar, A., Emaru, T. and Kobayashi, Y., 2020. HPPRM: hybrid potential based probabilistic roadmap algorithm for improved dynamic path planning of mobile robots. *IEEE Access*, 8, pp.221743-221766.

[19] Zhang Ke. Research on indoor robot mapping and navigation algorithm based on depth camera[D]. Harbin Institute of Technology, 2020.