Research on manipulation algorithm of HRDA based on visual UTFastSLAM

Huazhong Li*

Shenzhen Institute of Information Technology, Shenzhen, 518172, Guangdong, China

Abstract. Aiming at the **SLAM** problem faced by humanoid robot path planning in unknown environment, firstly, based on the analysis of feature region location and information description in the region, a feature extraction algorithm based on **LPP** has been researched to reduce the computational complexity of **SLAM** and the sensitivity to wrong data association. Secondly, aiming at the problems of phase winding and phase singularity in visual phase matching, a phase matching algorithm based on adaptive Gabor filter has been researched. By extracting the phase difference and visual difference of image pairs, there is no need to explicitly calculate the local phase, so as to achieve the best matching between the ideal phase difference and the world measured phase difference. Using unscented transform (**UT**) and selective acquisition strategy, a **UTFastSLAM** algorithm based on Rao-Blackwellised has been proposed, which effectively reduces the particle dilution phenomenon of particle filter (**PF**) and significantly improves the calculation accuracy and robustness of **FastSLAM** algorithm. Finally, the **HRDA** experimental platform has been constructed, and the forward and reverse kinematics models of **HRDA** have been established to verify the effectiveness of the algorithm proposed.

Keywords: **HRDA**, **SLAM**, Path planning, **LPP**, Unscented transform (**UT**).

1 Introduction

Simultaneous Localization and Mapping (**SLAM**) is the key technology of humanoid robot path planning. In recent years, with the improvement of the cost performance of embedded computers and the popularity of visual sensors, the realization of humanoid robot **SLAM** using visual information has gradually become a hot issue of general research concern by scholars at home and abroad [1-3]. The goal of visual **SLAM** is to recursively recover the position of camera motion and environmental features from visual information based on Bayesian filtering technology. The biggest advantage of visual information is abundant information and high cost performance. However, projection transformation makes depth information recovery a very difficult problem, and the three-dimensional position of feature points is very important for **SLAM**. Although the recovery of depth information can be realized by binocular or multi camera stereo camera, it increases the manufacturing cost. For

* Corresponding author: lihz@sziit.com.cn

monocular cameras, the restoration of depth information has to recover three-dimensional information from multiple two-dimensional images with the help of the motion between frames.

Davis [4] first carried out the research on monocular vision **SLAM**, and achieved some research results in indoor active vision **SLAM** and positioning, but its feature extraction and matching have strict requirements on the environment, such as there must be stable pane corners, and there must be no errors in matching. In addition, some other researchers have also proposed many new research methods and improvement means [5-7]. Previous studies have shown that feature extraction and matching are particularly important for the **SLAM** of monocular vision. Once robust and stable image features are obtained, motion restoration and depth restoration are feasible.

Robust invariant features have attracted extensive attention of researchers at home and abroad because of their important application value. Among them, invariant features that do not change with the viewpoint have been deeply studied, especially Locality Preserving Projections (**LPP**) features, which only use the information of the local region of the target to construct the feature quantity. Because these local regions may appear discretely in different positions of the target, when extracting features independently for each region, even if the target is located in a complex environment or partially occluded, some information of the target can be obtained through local feature extraction, so as to realize target recognition in complex background, which has a wider application. **LPP** extraction method has become a very active research direction [8-11], involving target recognition, panoramic image mosaic, virtual reality, robot autonomous navigation and other fields. For the application of extracted features in **SLAM**, we must also consider the problem of feature matching and management. Especially for large-scale or texture rich scenes, the feature accumulation speed is fast. On the one hand, it improves the accuracy of **SLAM** estimation, on the other hand, it also brings the problem of slow feature matching. Therefore, effective feature matching and feature management methods are necessary. In addition, in order to effectively reduce or eliminate the cumulative error generated in the motion process of humanoid robot, **SLAM** requires to estimate the consistent trajectory and environmental landmark state of humanoid robot. At present, **SLAM** solutions are mainly based on Bayesian filtering framework, and typical methods mainly include: Extended Kalman Filter (**EKF**) based on full state and Particle Filter (**PF**) [12]. Because the computational complexity of **EKF** is O(n2), it cannot be applied in large-scale environment. **PF** scheme decouples the whole state problem of **SLAM**, greatly reduces the computational complexity, and provides a more feasible method for solving large-scale environment. The fast Simultaneous Localization and Mapping (**FastSLAM**) based on PF proposed by Montemerlo et al. is the most representative [13-14]. At present, this method has received extensive attention in **SLAM**. However, the traditional **FastSLAM** method usually requires a large number of particles. Because each particle corresponds to a landmark map, if the number of particles is too large, it will greatly increase the computational burden and storage space, and **EKF** needs to be used for the update of each landmark map, so there are similar problems and limitations of **EKF**. Therefore, aiming at the **SLAM** problem faced by humanoid robot path planning in unknown environment, firstly, in order to reduce the computational complexity of **SLAM** and the sensitivity to wrong data association, a feature extraction algorithm based on **LPP** has been researched. Secondly, aiming at the problems of phase winding and phase singularity in visual phase matching, a phase matching algorithm based on adaptive Gabor filter has been proposed; In order to effectively reduce the particle dilution phenomenon of PF and improve the calculation accuracy and robustness of **FastSLAM** algorithm, the **UTFastSLAM** algorithm based on Rao-Blackwellised has been researched by using unscented transformation (**UT**) and selective acquisition strategy. Finally, the Humanoid Robot Dual Arms (**HRDA**)

experimental platform has been constructed, and the forward and reverse kinematics models of **HRDA** have been established to verify the effectiveness of the algorithm proposed.

2 Feature extraction algorithm based on LPP

The feature extraction algorithm based on Local Preserving Projections (**LPP**) mainly solves two problems: (1) Location of feature region, that is, where to extract features? (solve the where problem). (2) Description of information in the region, that is, what feature quantity is used to describe the information in the region? (solve the what problem). The so-called feature region location refers to the construction of multi-scale differential Gaussian space based on scale space theory, the use of automatic scale selection method to locate the location of key points and determine the size of the feature region. The so-called intra domain information description means that in order to make the extracted features have good singularity, a well differentiated and deterministic feature description operator must be designed, and it has strong adaptability to the illumination, displacement, vision and noise of the image. This paper studies a new descriptor, **LPP** feature descriptor. The so-called **LPP** feature is a point feature, which is the pole not near the image edge in three consecutive Gaussian difference images. The pole in the Gaussian difference image is the pixel whose gray value is larger or smaller than its surrounding pixels. It is the pixel with the most drastic change after Gaussian blur. Therefore, it has good stability. Using the extreme value of the convolution $D(x, y, \sigma)$ between the Gaussian difference function and the image space in the scale space. $D(x, y, \sigma)$ can be obtained from two adjacent Gaussian convolution images with a scale difference of a constant factor **K**, which can be expressed as formula (1).

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (1)$$

An effective way to build $D(x, y, \sigma)$ is described below. First, the original image is convoluted with the Gaussian function to generate the pre-processed blurred image I_0, and then I_0 is convoluted with the Gaussian function to generate the first blurred image I_1 of the first layer. Then, I_2 is obtained by convolution of I_1 and Gaussian function ($\sigma_2 = k\sigma_1$) and iterated successively. Secondly, divide each layer of the scale space into s intervals, then $k = 2^{1/s}$ .In the fuzzy image stack, s + 3 images must be generated for each layer, so that the later extreme point detection can cover the whole layer. Then, s + 2 Gaussian difference images are generated by subtracting the adjacent Gaussian blurred images.After processing the first layer, resample the third Gaussian image I_{s+1} at the top of the first layer stack. The resampling process is to scan the pixels of I_{s+1} in rows and columns, so as to obtain the first blurred image of the second layer, which is 1 / 4 of the size of the first layer. Then use the above method to generate the next layer of Gaussian blurred image and Gaussian difference image, and so on, Until the image is reduced to a certain extent. Therefore, multi-layer Gaussian blur and Gaussian difference images can be obtained, which are Gaussian blur images and Gaussian difference images. Because the local minima and maxima in the Gaussian difference image $D(x, y, \sigma)$ may have feature points with constant proportion. In order to detect these extreme points, $D(x, y, \sigma)$ is sampled first, and each sampling point is compared with 8 adjacent points of the current image and 9 adjacent points of the upper and lower adjacent scales. The condition for sampling points to be extracted is that their gray

values are greater than or less than all 26 adjacent points. The sampling points detected above are called candidate feature points. In order to obtain more accurate image coordinates of feature points, it is necessary to compare the sampling points with their surrounding pixels. On the one hand, it can make the data of sampling points (including position, scale, main curvature ratio, etc.) more accurate. On the other hand, it can also exclude those sampling points with small difference from adjacent points or poor quality at the edge of the image. This paper adopts the positioning method proposed by brown, which has good positioning accuracy. In this method, a three-dimensional quadratic function is used to fit the local sampling points to determine the interpolation position of the maximum value. The scale space function $D(x, y, \sigma)$ is expanded by second-order Taylor expansion, which can be expressed as formula (2).

$$D(x) = D + \partial D^T / \partial x + 1/2 \cdot x^T \, \partial^2 D / \partial^2 x \cdot x \tag{2}$$

where, D and its partial derivative are the values at the sampling point, and $x = (x, y, \sigma)$ is the offset relative to the sampling point. By calculating the partial derivative of the above function with respect to x and making it zero, the position $\hat{x}$ of the extreme point can be obtained, which can be expressed as formula (3).

$$\hat{x} = - \partial^2 D^{-1} / \partial^2 x \cdot \partial D / \partial x \tag{3}$$

Substituting $\hat{x}$ into equation (2) yields equation (4).

$$D(\hat{x}) = D + 1/2 \cdot \partial D^T / \partial x \cdot \hat{x} \tag{4}$$

In this experiment, all extreme points of $D(\hat{x}) < 0.02$ will be deleted. In this way, according to the function value at the extreme point $\hat{x}$, those unstable extreme points with small difference from the neighborhood points can be excluded. According to the gray level change of the local image around the feature points, each feature point is given a constant direction. In this way, the feature points can be described with the information related to the direction, so as to obtain the invariance of image rotation. Firstly, the Gaussian blurred image l with the scale closest to the feature point is selected so that all calculations can be carried out in a scale invariant manner. At this scale, the gradient value $m(x, y)$ and direction $\theta(x, y)$ of each image sampling point $L(x, y)$ are calculated by the difference between pixels, as shown in formula (5) below.

$$\begin{cases} m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \\ \theta(x,y) = tan^{-1}((L(x+1,y) - L(x-1,y))/(L(x,y+1) - L(x,y-1)) \end{cases} \tag{5}$$

According to the gradient direction of sampling points in a certain area around the feature points, a direction histogram can be established, which contains 36 heaps and covers the direction domain of 360 degrees. Each sampling point added to the histogram is weighted according to its gradient value and Gaussian weighted circular window.

Each peak in the direction histogram corresponds to the dominant direction of each local gradient. First, the highest peak is detected. Then, any other local peak above 80% of the highest peak here is also used to create a feature point in the corresponding direction; If there are multiple peaks with close gradient values at one location, multiple feature points with the

same location and scale but different directions will be created at that location; Only about 15% of points are given multiple directions, but these points play a great role in the stability of matching. Finally, for each peak, the stack values of the three histograms closest to the peak are used for parabolic interpolation fitting to make the position of the peak more accurate.

3 Binocular visual feature matching algorithm

The phase matching method used in this paper has the advantages of high visual accuracy, good stability and parallel computing. It is an important research method in binocular stereo matching. Aiming at the problems of phase winding and phase singularity in the existing phase matching methods and meeting the requirements of real-time image matching, this paper studies a binocular visual feature matching algorithm to extract the phase difference of image pairs based on adaptive Gabor filter, and then calculate the parallax. There is no need to explicitly calculate the local phase, which has high operation efficiency. For any given parallax, the task of stereo matching is transformed into finding a parallax value, so that the ideal phase difference and the actual measured phase difference can achieve the best matching. The algorithm can obtain accurate parallax estimation, has high robustness, and the predicted parallax range is large. In addition, RANdom SAmple Consensus (**RANSAC**) constraints, epipolar geometric constraints and uniqueness constraints are used to eliminate wrong matching pairs. Therefore, it is necessary to study the basic properties and time-frequency localization properties of wavelet. For the Fourier transform $\hat{f}(\varpi)$ of function $f(x)$, equation (6) is satisfied.

$$\hat{f}(\bullet - b) = e^{-ib\varpi}\hat{f}(\varpi), [\hat{f}(\bullet \ / a)](\varpi) = |a|\hat{f}(a\varpi) \tag{6}$$

Now we discuss the time-frequency localization property of the window function defined by equation (6). From the perspective of time domain, when $\psi_{a,b}(t)$ is used as a window function, the center t_0 and window width $\sigma_{\psi_{a,b}}$ can be expressed as follows respectively.

$$\mathbf{t_0} = \frac{\mathbf{1}}{\left\|\boldsymbol{\psi}_{\mathbf{a,b}}\right\|_2^2}\int_{\mathbf{R}} \mathbf{t}\left|\boldsymbol{\psi}_{\mathbf{a,b}}(\mathbf{t})\right|^2 \mathbf{dt}, \quad \boldsymbol{\sigma}_{\boldsymbol{\psi}_{\mathbf{a,b}}} = \frac{\mathbf{1}}{\left\|\boldsymbol{\psi}_{\mathbf{a,b}}\right\|_2}\left\{\int_{\mathbf{R}} (\mathbf{t}-\mathbf{t_0})^2\left|\boldsymbol{\psi}_{\mathbf{a,b}}(\mathbf{t})\right|^2 \mathbf{dt}\right\}^{1/2} \tag{7}$$

Equation (7) can be expressed in the form of convolution $f * \psi((\bullet - b)/a)$. From the perspective of frequency, there is $W_\psi f(a,b) = \int_{-\infty}^{+\infty}\hat{f}(\varpi)\hat{\psi}_{a,b}(\varpi)d\varpi \Big/ (2p)$ by using Parseval energy conservation theorem. Therefore, using $\hat{\psi}_{a,b} = |a|^{1/2} e^{-ib\varpi}\hat{\psi}(a\varpi)$, the center ϖ and width $\sigma_{\psi_{a,b}}$ of the frequency window can be expressed as follows.

$$\varpi_0 = \frac{1}{\left\|\hat{\psi}_{a,b}\right\|_2^2}\int_R \varpi\left|\hat{\psi}_{a,b}(\varpi)\right|^2 d\varpi, \quad \sigma_{\psi_{q,b}} = \frac{1}{\left\|\hat{\psi}_{a,b}\right\|_2}\left\{\int_R (\varpi-\varpi_0)^2\left|\hat{\psi}_{a,b}(\varpi)\right|^2 d\varpi\right\}^{1/2} \tag{8}$$

Note $\psi_{a,b}(t)=\psi(t)$ in $a=1$ and $b=0$, i.e. $\psi_{a,b}(t)|_{a=1,b=0}=\psi(t)$, and the corresponding time domain and frequency domain window parameters are recorded as t_0^*, σ_ψ, ϖ_0^* and $\sigma_{\hat{\psi}}$ respectively. Equation (9) can be established.

$$t_0=at_0^*+b,\sigma_{\psi_{a,b}}=a\sigma_\psi,\varpi=\varpi_0^*/a,\sigma_{\hat{\psi}_{a,b}}=\sigma_{\hat{\psi}}/a \tag{9}$$

If calculated directly, equation (10) can be obtained.

$$t_0=\int_R t\left|\frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right)\right|^2 dt\Bigg/\int_R\left|\frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right)\right|^2 dt=at_0^*+b \tag{10}$$

Similarly, equation (11) is obtained.

$$\sigma_{\psi_{a,b}}=a\sigma_\psi \tag{11}$$

The time-frequency window satisfies the following equation (12).

$$\left[t_0-\sigma_{\psi_{a,b}},t_0+\sigma_{\psi_{a,b}}\right]\times\left[\varpi_0-\sigma_{\psi_{a,b}},\varpi_0+\sigma_{\psi_{a,b}}\right]=\left[at_0^*+b-a\sigma_\psi,at_0^*+b+a\sigma_\psi\right]\times\left[\frac{\varpi_0-\sigma_{\hat{\psi}}}{a},\frac{\varpi_0+\sigma_{\hat{\psi}}}{a}\right] \tag{12}$$

At this time, the time width of the window is $2a\sigma_\psi$ and the frequency width is $2\sigma_{\hat{\psi}}/a$. Therefore, its area is $4\sigma_\psi\times\sigma_{\hat{\psi}}$, which is independent of the selection of parameters a and b. The time-frequency window has the following characteristics: (1) when the high-frequency component needs to be detected, reduce the value of a, at this time, the time window automatically narrows, and the frequency window automatically widens. At this time, it is a high-frequency window with a narrow width and a large frequency width; (2) When detecting the low-frequency component, increase the value of a, the time window will be automatically wide, and the frequency window will be automatically narrow. At this time, it is a low-frequency window with large time and narrow frequency width.

Next, the physical significance of wavelet transform is discussed from the perspective of system response. Set the input information as $f(x)$ and the unit excitation response of the system as $h_a(t)=h(-t/a)/\sqrt{|a|}$, so the output of the system satisfies equation (13).

$$f(\bullet)\otimes h_a(\bullet)=\int_R f(x)\psi((x-b)/a)\Big/\sqrt{|a|}\cdot dx=W_\psi f(a,b) \tag{13}$$

Therefore, the continuous wavelet transform of $f(x)$ can also be regarded as the transfer function and the output of $H_a(\omega)$ system. In addition, through the time-frequency analysis of the previous window function, it can be seen that $\psi(t)$ is essentially a band-pass system, and with the change of scaling factor a. $\psi_{a,b}(t)$ corresponds to a series of bandpass systems with different bandwidth and center frequency. The following physical characteristics of wavelet transform are summarized.

(1) The continuous wavelet transform of signal $f(x)$ is the output of $f(x)$ filtered by a series of band-pass filters. Parameter a in $W_\psi f(a,b)$ reflects the bandwidth and center frequency of the band-pass filter, while parameter b reflects the time parameter of the filtered output.

(2) Let Q be the ratio of the center frequency and bandwidth of the filter, that is, the quality factor, then the band-pass filters formed by the change of scaling factor a are constant Q filters.

(3) When the scaling factor a changes, the bandwidth and center frequency of the bandpass filter also change. When parameter a is smaller, the center frequency is larger and the bandwidth becomes wider; When parameter a is large, the center frequency is small and the bandwidth becomes narrow. This characteristic of wavelet transform has important application value for the local characteristic analysis of signal $f(x)$. For example, where the signal changes slowly, it is mainly low-frequency components and the frequency range is relatively narrow. At this time, the band-pass filter of wavelet transform is equivalent to a large case; On the contrary, where the signal changes suddenly, it is mainly high-frequency components, and the frequency range is relatively wide. The band-pass filter of wavelet transform is equivalent to a small case. In short, when the scaling factor changes from small to large, the filtering range changes from high frequency to low frequency. Therefore, wavelet transform has zoom characteristics.

(4) Linear transformation. Wavelet transform is a linear transform, which has the property of linear transform: superposition.

(5) Time shift characteristics.

$$\text{if } \mathrm{f(x)} \xrightarrow{\text{CWT}} \mathrm{W_f(a,b)} \text{ ,then } \mathrm{f(x-x_0)} \xrightarrow{\text{CWT}} \mathrm{W_f(a,b-x_0)} \tag{14}$$

(6) Scale conversion.

$$\text{if } \mathrm{f(x)} \xrightarrow{\text{CWT}} \mathrm{W_f(a,b)} \text{ ,then } \mathrm{f(x/\lambda)} \xrightarrow{\text{CWT}} \sqrt{\lambda}\,\mathrm{W_f(a/\lambda,b)} \tag{15}$$

(7) Redundancy. The redundancy of wavelet transform is illustrated by the reconstruction kernel and the reconstruction kernel equation. That is, the wavelet transform values of each point in the (a,b) half plane are related. The wavelet transform value at point (a_0,b_0) can be represented by the wavelet transform value of each point in the (a,b) half plane.

$$\begin{cases} \mathrm{W_f(a_0,b_0)} = \int_0^{+\infty} \dfrac{\mathrm{da}}{\mathrm{a^2}} \int_{-\infty}^{+\infty} \mathrm{W_f(a_0,b_0)\,K_\psi(a_0,b_0,a_0,b_0)\,d\tau} \\ \mathrm{K_\psi(a_0,b_0,a_0,b_0)} = \dfrac{1}{\mathrm{c_\psi}} \int \psi_{\mathrm{a,b}}(\mathrm{x})\,\psi^*_{\mathrm{a_0,b_0}}(\mathrm{x})\,\mathrm{dx} = \dfrac{1}{\mathrm{c_\psi}} \left\langle \psi_{\mathrm{a,b}}(\mathrm{x}), \psi_{\mathrm{a_0,b_0}}(\mathrm{x}) \right\rangle \end{cases} \tag{16}$$

where, K_ψ is the inner product of $\psi_{a,b}(x)$ and $\psi_{a_0,b_0}(x)$, which reflects the correlation between them. It is called reconstructed kernel. When $a=a_0$ and $a=b_0$, K_ψ has the maximum value. When (a,b) deviates from (a_0,b_0), the value of K_ψ decays rapidly, and the smaller the correlation region between them. If $K_\psi = d(a-a_0,b-b_0)$, at this time, the wavelet transform values in (a,b) plane are not related to each other, and the

information contained in wavelet transform is not redundant, which requires that wavelets with different scales and different translations are orthogonal to each other. However, when (a,b) is a continuous variable, it is difficult to meet this requirement.

(8) Properties of cross terms of wavelet spectra. Wavelet transform is linear and there is no cross term. However, the energy distribution function $\left|W_f(a,b)\right|^2$ derived from wavelet transform has cross terms in the case of multiple signals. If $f(t)=f_1(t)+f_2(t)$ is set, equation (17) can be obtained.

$$\left|W_f(a,b)\right|^2 = \left|W_{f_1}(a,b)\right|^2 + \left|W_{f_2}(a,b)\right|^2 + 2\left|W_{f_1}(a,b)\right| \bullet \left|W_{f_2}(a,b)\right| \bullet \cos(\theta_1-\theta_2) \quad (17)$$

The cross term of wavelet transform only appears at (a,b) where W_{f_1} and W_{f_2} are not zero at the same time. This is different from Wigner distribution.

4 UTFastSLAM algorithm based on rao-blackwellised

Unbiased Transform (**UT**) is a new method to calculate the statistical characteristics of nonlinear transformation random variables. It has strong advantages in solving nonlinear and non-Gaussian problems. Because the environment perception function is nonlinear due to the complexity of the environment when creating the indoor environment map, the linear approximation of **EKF** in the classical algorithm may produce large errors. In this paper, **UT** is introduced into **FastSLAM** to reliably approximate the posterior distribution of the pose of the humanoid robot, which overcomes the loss problem of the traditional particle method, and greatly reduces the number of particles through effective adaptive resampling, so that a good approximate posterior distribution can be obtained with only a few particles. Firstly, the current perceptual information z_t is added to the traditional proposed distribution $p(s_t \mid s_{1:t-1}^{(i)}, u_{1:t-1}, z_t)$, so that the particles sampled from the prior distribution move to the posterior high probability region. The traditional method to approximate the probability is to use the Gaussian approximation generated by **EKF**, which can be expressed as formula (18).

$$p(s_t \mid s_{1:t-1}^{(i)}, u_{1:t-1}, z_t) \approx N(s_t, \bar{s}_t, P_{s_t}) \quad (18)$$

EKF is approximately obtained by the first-order Taylor expansion of the nonlinear perception function $z_t = g(M, s_t)$ in the mean $\bar{s}_t$, which can be expressed as equation (19).

$$z_t = g(M,s_t) \approx g(M,\bar{s}_t) + \Delta_{s_t} g'(M,\bar{s}_t) \quad (19)$$

Therefore, the mean and variance of the perceptual variable $z_t = g(M,s_t)$ can be expressed as equation (20) respectively.

$$\bar{z}_t = g(M,\bar{s}_t), P_{z_t} = g'(M,\bar{s}_t)^T g'(M,\bar{s}_t) \quad (20)$$

In this paper, **UT** is used to calculate the mean $\bar{z}_t$ and variance P_{z_t} of perceptual

variable z_t, and its accuracy can reach the second-order Taylor expansion. Let L be the dimension of s_t, which is mainly divided into the following steps.

Step 1 Generate $2L+1$ sigma points $s_i = \{x_i, y_i\}$ and obtain formula (21) ~ (24).

$$x_0 = \bar{s}_t, x_i = \bar{s}_t + (\sqrt{(L+\lambda)P_{s_t}})_i, i = 1, \cdots, L \tag{21}$$

$$x_i = \bar{s}_t - (\sqrt{(L+\lambda)P_{s_t}})_i, i = L+1, \cdots, 2L \tag{22}$$

$$w_0^m = \lambda/(L+\lambda), w_0^c = w_0^m + (1-\alpha^2+\beta) \tag{23}$$

$$w_i^m = w_i^c = 1/2(L+\lambda), \lambda = \alpha^2(L+\gamma) - L, i = 1, \cdots, 2L \tag{24}$$

Where, γ is a scale parameter and controls the distance between the sigma point and the mean $\bar{s}_t$. α is a positive definite scale parameter that controls the higher-order effects produced by the nonlinear function $g(\bullet)$. β is a parameter that controls the weight of the 0th sigma point. $\alpha = 0$, $\beta = 0$ and $\gamma = 2$ are commonly used optimal values. It should be noted that the weight of the 0th sigma point is different when calculating the mean and covariance, which are w_0^m and w_0^c respectively.

Step 2 The sampling y_i of the random variable y is calculated according to the nonlinear function $g(\bullet)$, which can be expressed as equation (25).

$$y_i = g(x_i), \ i = 0, \cdots, 2L \tag{25}$$

Step 3 Calculate the mean $\bar{y}$ and covariance P_{yy} of the random variable y, which can be expressed as equation (26).

$$\bar{y} = \sum_{i=0}^{2L} w_i^m y_i, P_{yy} = \sum_{i=0}^{2L} w_i^c (y_i - \bar{y}_i)(y_i - \bar{y}_i)^T \tag{26}$$

Secondly, the humanoid robot path $s_{1:t}^{(i)}$ can be extended by sampling the new pose $s_t^{(i)}$ from the proposed distribution $p(s_t \mid s_{1:t-1}^{(i)}, u_{1:t-1}, z_t)$ absorbing the current perception z_t according to the UT algorithm, and the steps are as follows.

Step 1 For the i-th pose estimation $\mathrm{st}-1(\mathrm{i})$ of a priori random variable $\mathrm{st}-1$, calculate 2L+1 sigma points $\left\{x_{t-1}^{(i),(j)}, w_{t-1}^{(i),(j)}\right\}$ ($i = 1, \cdots, N$ is the number of particles and $j = 1, \cdots, 2L+1$ is the number of sigma points).Step 2 Apply motion model prediction. The input information is divided into two categories: Based on odometer reading and based on scan matching results. Since the odometer reading translation $d_t'^{(i)}$ and deflection $\alpha_t'^{(i)}$ are relatively reliable in a small range, at this time, the input information $u_{t-1}^{(i)} = \left\{d_t'^{(i)}, \ \alpha_t'^{(i)}\right\}$ of the motion model $p(s_t^{(i)} \mid s_{t-1}^{(i)}, u_{t-1}^{(i)})$ can be expressed as equation (27) based on the prediction of the motion model.

$$\begin{cases} x_{t|t-1}^{*,(i),(j)} = f(x_{t-1}^{(i),(j)}, u_{t-1}^{(i)}), \bar{s}_{t|t-1}^{(i)} = \sum_{j=0}^{2L} w_j^{m,(i)} x_{t|t-1}^{*,(i),(j)} \\ P_{t|t-1}^{(i)} = \sum_{j=0}^{2L} w_j^{c,(i)} [x_{t|t-1}^{*,(i),(j)} - \bar{s}_{t|t-1}^{(i)}][x_{t|t-1}^{*,(i),(j)} - \bar{s}_{t|t-1}^{(i)}]^T \end{cases} \tag{27}$$

Where, the motion model $p(s_t^{(i)} \mid s_{t-1}^{(i)}, u_{t-1}^{(i)})$ of the humanoid robot is represented by $f(_{t-1}^{(i)}, u_{t-1}^{(i)})$, and $x_{t-1}^{(i),(j)} = [x_{t-1}^{(i),(j)}, y_{t-1}^{(i),(j)}, \theta_{t-1}^{(i),(j)}]^T$ represents the posture of the humanoid robot at the j sigma point of the i-th particle at $t-1$. The motion model describes that the pose (position and pose) of the humanoid robot at time $t-1$ is $s_{t-1}^{(i)}$. Under the condition that the input information is $u_t^{(i)}$, the probability distribution of the pose of the humanoid robot at time t. Generally, the following motion model can be expressed as equation (28).

$$x_t = x_{t-1} + d_t \cos(\theta_{t-1} + \alpha_t), y_t = y_{t-1} + d_t \sin(\theta_{t-1} + \alpha_t), \theta_t = \theta_{t-1} + \mathrm{mod}(\alpha_t, 2\pi) \tag{28}$$

Step 3 Absorbing the current perceived information z_t can be expressed as equation (29).

$$\begin{cases} Z_{t|t-1}^{*,(i),(j)} = g(x_{t|t-1}^{*,(i),(j)}, M), \bar{z}_{t|t-1}^{(i)} = \sum_{j=0}^{2L} w_j^{m,(i)} Z_{t|t-1}^{*,(i),(j)} \\ P_{z_t z_t}^{(i)} = \sum_{j=0}^{2L} w_j^{c,(i)} [Z_{t|t-1}^{*,(i),(j)} - \bar{z}_{t|t-1}^{(i)}][Z_{t|t-1}^{*,(i),(j)} - \bar{z}_{t|t-1}^{(i)}]^T \\ P_{s_t z_t}^{(i)} = \sum_{j=0}^{2L} w_j^{c,(i)} [x_{t|t-1}^{*,(i),(j)} - \bar{s}_{t|t-1}^{(i)}][Z_{t|t-1}^{*,(i),(j)} - \bar{z}_{t|t-1}^{(i)}]^T \\ \bar{s}_t^{(i)} = \bar{s}_{t|t-1}^{(i)} = K_t^{(i)}(z_t^{(i)} - z_{t|t-1}^{(i)}), K_t^{(i)} = P_{s_t z_t}^{(i)} (P_{z_t z_t}^{(i)})^{-1}, P_t^{(i)} = P_{t|t-1}^{(i)} - K_i^{(i)} P_{z_t z_t}^{(i)} K_i^{(i)T} \end{cases} \tag{29}$$

The new pose $s_t^{(i)}$ of the humanoid robot is sampled and the humanoid robot path $s_{1:t}^{(i)}$ is extended, which can be expressed as equation (30).

$$s_t^{(i)} \sim p(s_t \mid s_{1:t-1}^{(i)}, u_{1:t-1}, z_t) = N(s_t; \bar{s}_t^{(i)}, P_t^{(i)}), s_{1:t}^{(i)} = (s_{1:t-1}^{(i)}, s_t^{(i)}) \tag{30}$$

When the signpost is updated, a posteriori estimation of the i-th particle and the k_t signpost $m_{k_t,t-1}^{(i)} = \left\{\mu_{k,t-1}^{(i)}, \Sigma_{k_t,t-1}^{(i)}\right\}$ is realized. The updated value $\left\{\mu_{k_t,t}^{(i)}, \Sigma_{k_t,t}^{(i)}\right\}$ is added to the temporary particle set **St** together with the new humanoid robot sampling pose st(i). The update of the landmark k_t depends on whether it is perceived at time t , that is, whether the scanning data obtained at time t is successfully matched with the scanning data related to the landmark k_t in the map library. If the matching is successful, it will be updated, otherwise it will not be perceived, and the pose will remain unchanged. The update formula can be expressed as equation (31).

$$p(m_{k_t,t}^{(i)} \mid z_k^{(i)}, s_k^{(i)}, k_t) = \eta \underbrace{p(z_t^{(i)} \mid m_{k_t,t}^{(i)}, s_t^{(i)}, z_{1:t-1}^{(i)}, k_t)}_{\sim N(z_t; g(m_{k_t,t}^{(i)}, s_t^{(i)}), R_t)} \underbrace{p(m_{k_t,t}^{(i)} \mid s_t^{(i)}, z_{1:t-1}^{(i)}, k_t)}_{\sim N(m_{k_t,t}^{(i)}; \mu_{k_t,t-1}^{(i)}, \Sigma_{k_t,t-1}^{(i)})} \tag{31}$$

Where, the probability $p(m_{k_t,t}^{(i)} \mid z_{t-1}^{(i)}, s_{t-1}^{(i)}, k_t)$ at time $t-1$ is expressed as Gaussian

distribution $N(\mu_{k_t,t-1}^{(i)}, \Sigma_{k_t,t-1}^{(i)})$. The new estimation of time t is also Gaussian. It is necessary to generate Gaussian approximation to the perception model, and use **UT** method to approximate the nonlinear perception function $g(m_{k_t,t}^{(i)}, s_t^{(i)})$ .Generate 2L + 1 sigma points $(\xi_{k_t,t-1}^{(i),(j)}, w_{k_t,t-1}^{(i),(j)})$, $j = 0,\cdots,2L$, which can be expressed as equation (32).

$$\begin{cases} \xi_{k_t,t-1}^{(i),(0)} = \mu_{k_t,t-1}^{(i)}, \xi_{k_t,t-1}^{(i),(j)} = \mu_{k_t,t-1}^{(i)} + (\sqrt{(L+\lambda)\Sigma_{k_t,t-1}^{(i)}})_j, j = 1,\cdots,L \\ \xi_{k_t,t-1}^{(i),(j)} = \mu_{k_t,t-1}^{(i)} - (\sqrt{(L+\lambda)\Sigma_{k_t,t-1}^{(i)}})_j, j = L+1,\cdots,2L \\ w_{k_t,t-1}^{\prime(i),(0)} = \lambda/(L+\lambda), w_{k_t,t-1}^{\prime\prime(i),(0)} = w_{k_t,t-1}^{\prime(i),(0)} + (1-\alpha^2+\beta) \\ w_{k_t,t-1}^{\prime(i),(j)} = w_{k_t,t-1}^{\prime\prime(i),(j)} = 1/2(L+\lambda), \lambda = \alpha^2(L+\lambda) - L, j = 1,\cdots,2L \end{cases} \tag{32}$$

The mean $z_{k_t,t}^{(i),(j)}$ and covariance $Q_{k_t,t}^{(i)}$ of perception calculated by perception model can be expressed as equation (33).

$$z_{k_t,t}^{(i),(j)} = g(\xi_{k_t,t-1}^{(i),(j)} - s_t^{(i)}), \bar{z}_{k_t,t}^{(i)} = \sum_{j=0}^{2L} w_{k_t,t-1}^{\prime(i),(j)} z_{k_t,t}^{(i),(j)}, Q_{k_t,t}^{(i)} = \sum_{j=0}^{2L} w_{k_t,t-1}^{\prime\prime(i),(j)} [z_{k_t,t}^{(i)} - \bar{z}_{k_t,t}^{(i)}][z_{k_t,t}^{(i)} - \bar{z}_{k_t,t}^{(i)}]^T \tag{33}$$

According to this approximation, the posteriori of the pose of the landmark $m_{k_t,t}^{(i)}$ is Gaussian distribution, and the mean $\mu_{k_t,t}^{(i)}$ and variance $K_t^{(i)}$ of the landmark are updated according to the update rules of formula (34) below.

$$\begin{cases} \mu_{k_t,t}^{(i)} = \mu_{k_t,t-1}^{(i)} + K_t^{(i)}(z_t^{(i)} - \bar{z}_t^{(i)})^T, \Sigma_{k_t,t}^{(i)} = (I - K_t^{(i)} Q_{k_t,t}^{(i)T})\Sigma_{k_t,t-1}^{(i)} \\ K_t^{(i)} = \Sigma_{k_t,t-1}^{(i)} Q_{k_t,t}^{(i)} (Q_{k_t,t}^{(i)T} \Sigma_{k_t,t-1}^{(i)} Q_{k_t,t}^{(i)} + R_t)^{-1} \end{cases} \tag{34}$$

The particles sampled from the proposed distribution may not well approximate the posterior distribution, and the difference between them needs to be measured by an evaluation. Therefore, the definition of weight coefficient can be expressed as equation (35).

$$w_t^{(i)} = \text{(target distribution)/ (proposed distribution)} \tag{35}$$

Where, the target distribution of the above formula is the posterior distribution $p(s_{1:t}^{(i)} \mid z_{1:t}, u_{1:t}, n_{1:t})$ of the humanoid robot path to be approximated by the sampled particles. Assuming that the path $s_{1:t}^{(i)}$ at the previous time is generated according to the posterior distribution $p(s_{1:t-1}^{(i)} \mid z_{1:t-1}, u_{1:t-1}, n_{1:t-1})$, the weight of the particles is derived according to the following formula (36).

$$\begin{aligned} w_t^{(i)} &= \frac{p(s_{1:t}^{(i)} \mid z_{1:t}, u_{1:t-1}, k_{1:t})}{p(s_{1:t-1}^{(i)} \mid z_{1:t-1}, u_{1:t-2}, k_{1:t-1}) p(s_t^{(i)} \mid s_{1:t-1}^{(i)}, u_{1:t-1}, z_{1:t}, k_{1:t})} \\ &\overset{Markov}{=} \eta \iint \underbrace{p(z_t \mid m_{k_t}, s_t, k_t)}_{\sim N(z_t; g(m_{k_t}, s_t), R_t)} \underbrace{p(m_{k_t} \mid s_{1:t-1}^{(i)}, z_{1:t-1}, u_{1:t-2}, k_{1:t-1})}_{\sim N(m_{k_t}, u_{k_t,t-1}^{(i)}, \Sigma_{k_t,t-1}^{(i)})} dm_{k_t} \underbrace{p(s_t \mid s_{1:t-1}^{(i)}, z_{t-1}, u_{t-1})}_{\sim N(s_t, \bar{s}_{t-1}^{(i)}, P_t)} ds_t \end{aligned} \tag{36}$$

The above formula can be obtained by the linear approximation of the nonlinear perception function g in the perception z_t, and the mean $\bar{z}_t^{(i)}$ and variance Q_t can be obtained directly through **UT**, then the weight approximation of the ith particle can be expressed as formula (37).

$$w_t^{(i)} = \left|2\pi L_t^{(i)}\right|^{\frac{1}{2}} \exp\left\{-\frac{1}{2}(z_t - \bar{z}_t^{(i)})^T L_t^{(i)-1}(z_t - \bar{z}_t^{(i)})\right\}, L_t^{(i)} = Q_t P_t Q_t^T + Q_t \sum\nolimits_{k_t,t-1}^{(i)} Q_t^T + R_t \tag{37}$$

Resampling can have a great impact on the performance of **PF**: not only low weight particles are replaced by high weight particles, but also only a limited number of necessary particles are allowed to approximate posteriori. Therefore, when the proposed distribution is quite different from the posterior distribution, resampling is very important, but resampling may also ignore some particles with high weight in the particle set, which can lead to filter divergence at worst. Since the new position particles in the **FastSLAM PF** are extracted from the proposed distribution motion model. If the shape of the proposed distribution is similar to that of the actual posterior distribution, the particles extracted according to the proposed distribution can well represent the posterior distribution after compensation by using the weight function. However, because the proposed distribution does not consider the current perceptual information, if the posterior distribution is located, there is a great difference between the proposed distribution and the actual posterior distribution, resulting in the divergence of **PF**, that is, the loss problem. That is, after several iterations, the weight of most particles tends to zero, so only a few particles really play a role in the estimation of the system state. Good particles may be ignored, and the particles will be concentrated in a small area. The proposed distribution designed in this paper integrates the current perception information and samples particles from the proposed distribution through **UT**, so that the particles move in the high probability region of the posterior distribution. Therefore, the particle distribution is well approximate to the posterior distribution, and **UKF** is used to update the road signs. For the case of nonlinear and non-Gaussian distribution, to avoid the problems caused by the linearization of nonlinear equations by **EKF**, while **UKF** uses a set of carefully selected weighted sampling points to express the statistical characteristics of the system. These samples evolve according to the real nonlinear equations without linearization. It is superior to **EKF** both in theory and in practical application.

5 Building HRDA experimental platform

Finally, this paper designs a specific experimental system to complete the design experiment. The humanoid robot is equipped with odometer and binocular vision sensor, and its CPU is an embedded multi-core processor. The function of the experimental software system is based on sensor information processing, and its main functional modules are described as follows.

(1) Wireless communication module. Provide information interaction between humanoid robot and the outside world and man-machine.

(2) Map creation module. The **UTFastSLAM** algorithm proposed in this paper is used for map creation.

(3) Locate the module. The pose of humanoid robot is determined according to the environmental perception information and the created map, which provides necessary information for the behavior decision of humanoid robot.

(4) Navigation module. On the created map, according to the current environmental perception information and the positioning information of humanoid robot, the functions of robot path planning and dynamic obstacle avoidance are realized in real time.

(5) Behavioral decision module. The complex robot makes accurate decisions according to the owner's instructions, the pose of the humanoid robot, the environmental map and the environmental perception information, controls the movement of the humanoid robot, and ensures that it can complete a certain action and the task assigned by the owner.

(6) Dual arm control module. On the basis of solving the environment map construction and object recognition of humanoid robot based on visual **UTFastSLAM** algorithm, the configuration space model, **WTRs**, forward and reverse kinematics of **HRDA** are established. Double arm collision avoidance and control **WTRs** algorithm based on **DLS-SVD** method and **BiRRT** to complete the control function of complex objects.

(7) Main program module. The main program module is responsible for the coordination and information exchange between various modules, and transmits the environmental perception information and human control information to other modules. The environment for binocular SLAM experiment adopts multiple rooms and corridors in the experimental building. There are computer tables, bookcases and other objects in the rooms. When the humanoid robot passes through the corridor, it may be dynamically disturbed by pedestrians. The tasks of global positioning, autonomous navigation and dual arm manipulation are realized on the created map.

5.1 Vision based global positioning

Through the robust feature extraction and matching algorithm proposed in this paper, combined with the multi vision geometric relationship in computer vision, the three-dimensional reconstruction algorithm is applied to solve the basic matrix and essential matrix of the projection equation, so as to obtain the relative pose between the humanoid robot and the recognition road sign, and then the humanoid robot can be located globally according to the absolute pose of the recognition road sign on the map. For the more challenging problem of humanoid robot abduction, because the posterior probability distribution of humanoid robot posture is a multi-peak distribution, it is necessary to reposition based on multi hypothesis tracking algorithm according to multiple observation information, and adopt the multi hypothesis tracking algorithm of central differential particle filter based on Gaussian mixture model, It can make the multi hypothesis posterior distribution of the pose of the humanoid robot quickly converge to the real pose of the humanoid robot, and solve the problem of relocation after abduction.

5.2 Vision based autonomous navigation

Vision based autonomous navigation can adopt vision measurement method and sub map matching method. The implementation process of visual measurement method is analysed. Through robust feature extraction and matching, the pose of humanoid robot is updated in time to improve the positioning accuracy. Effective visual servo control rules are applied to restrict the behavior of humanoid robot and ensure the real-time continuous navigation of humanoid robot. Aiming at the problem of error accumulation, a sub map matching method based on multi video frames with less computation is studied to update the pose of humanoid robot in real time.

5.3 HRDA operation task

The motion mechanism of **HRDA** used in this experiment is similar to the structure of human upper limbs and arms, which is mainly composed of trunk and left (right) arm. The trunk has three degrees of freedom. The joint motion $\boldsymbol{\theta}^t = [\varphi^t, \theta^t, \psi^t]^T$ is defined with reference to

the Euler angle **RPY** (the right superscript t represents the torso), which is the **Roll**(left and right swing around the R axis, abbreviated as $\mathbf{R_R}$), **Pitch**(back and forth bending and rotation around the P axis, abbreviated as $\mathbf{R_P}$) and **Yaw**(left and right rotation around the Y axis, abbreviated as $\mathbf{R_Y}$); The left (right) arm has seven degrees of freedom $\boldsymbol{\theta}^k = [\theta_1^k, \theta_2^k, \cdots, \theta_7^k]^T$ (the right superscript $\mathrm{k} = l, r$ represents the left arm and right arm , that is, $\boldsymbol{\theta}^l$ and $\boldsymbol{\theta}^r$ represent the configuration of the left and right arms respectively, which is similar below). They are three rotations of the shoulder joint ($\mathbf{R_P}$, $\mathbf{R_R}$ and $\mathbf{R_Y}$), one rotation of the axial joint around the P axis ($\mathbf{R_P}$) and three rotations of the wrist joint ($\mathbf{R_P}$, $\mathbf{R_R}$ and $\mathbf{R_Y}$). The three joints of the wrist always intersect at one point. The vector describing the terminal pose configuration of HRDA has 17 dimensions, which can be defined as $\boldsymbol{\Theta} = \left[[\boldsymbol{\theta}^t]^T, \ [\boldsymbol{\theta}^l]^T, \ [\boldsymbol{\theta}^r]^T\right]^T$. The vector describing the end pose configuration of the left (right) arm of the humanoid robot can be defined as $\boldsymbol{\Theta}^k = \left[[\boldsymbol{\theta}^t]^T, \ [\boldsymbol{\theta}^k]^T\right]^T$ .The origin o_w of the world coordinate system $\sum_w$ is located at the center point of the two feet. The origin of torso coordinate system $\sum_t$ is located at the center of the buttocks. The $\sum_{CoA}$ origin of the double shoulder center coordinate system is located at the center of the two shoulders. The distance between the shoulders is 2D, the trunk height is H_t and the lower limb height is H_f . Each coordinate system of the left (right) arm is defined as $\sum_i^k (o_i^k, x_i^k, y_i^k, z_i^k)$ (i=0,1,…,7).

*5.3.1 Building **HRDA** forward kinematics model*

HRDA direct kinematics (**DK**) is defined as: given the geometric parameters of each link member of humanoid robot and the configuration variables connecting each link member, find the pose of **HRDA** end effector relative to a specific coordinate system. In this paper, **Denavit-Hartenberg (D-H**) method is used to determine the pose of **HRDA**, establish its kinematic diagram, and form the family tree structure diagram of **HRDA**. Its **D-H** parameter is ($\theta_i, \alpha_i, a_i, d_i$).The **D-H** method is used to establish the homogeneous transformation matrix ${}^{i-1}\mathbf{T}_i$ of 4×4 for the connecting link coordinate system $\sum_i$ at each joint of **HRDA**, and describe its spatial relationship with the previous connecting link coordinate system $\sum_{i-1}$. Since all joints of **HRDA** are rotating joints (three rotating joints of trunk and seven rotating joints of left and right arms respectively), the general formula of ${}^{i-1}\mathbf{T}_i$ can be expressed as formula (38).

$$
\begin{aligned}
{}^{i-1}\mathbf{T}_i &= \mathbf{R}_{rot}(z_{i-1}, \theta_i) \cdot \mathbf{T}_{trans}(z_{i-1}, d_i) \cdot \mathbf{T}_{trans}(x_i, a_i) \cdot \mathbf{R}_{rot}(x_i, \alpha_i) \\
&= \begin{bmatrix} c_i & -c_{\alpha i} s_i & s_{\alpha i} s_i & a_i c_i \\ s_i & c_{\alpha i} c_i & -s_{\alpha i} c_i & a_i s_i \\ 0 & s_{\alpha i} & c_{\alpha i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned} \tag{38}
$$

According to the classical chain multiplication rule in the kinematics of humanoid robot, the kinematics equation in the form of **HRDA** forward homogeneous coordinate transformation from $\sum_w$ to $\sum_7^l$ （$\sum_7^r$） at the end of the left (right) arm can be expressed as equation (39).

$$ {}^{w}\mathbf{T}_7^k(\mathbf{\Theta}^k)={}^{w}\mathbf{T}_t\cdot{}^{t}\mathbf{T}_{CoA}(\boldsymbol{\theta}^t)\cdot{}^{CoA}\mathbf{T}_0^k\cdot{}^{0}\mathbf{T}_7^k(\boldsymbol{\theta}^k) \tag{39}$$

In equation (39), ${}^{w}\mathbf{T}_t=\begin{bmatrix}\mathbf{1}_{3\times3} & {}^{w}\mathbf{p}_t\\ \mathbf{0}_{1\times3} & 0\end{bmatrix}$ is the reversible constant homogeneous transformation matrix of $\sum_t$ relative to $\sum_w$. $\mathbf{1}_{3\times3}$ is the unit diagonal rotation matrix. $\mathbf{0}_{1\times3}$ is a zero matrix. ${}^{w}\mathbf{p}_t=\begin{bmatrix}0 & 0 & H_f\end{bmatrix}^T$ constant position vector. ${}^{t}\mathbf{T}_{CoA}(\boldsymbol{\theta}^t)$ is the reversible torso transformation matrix of $\sum_{CoA}$ relative to $\sum_t$, which can be expressed as equation (40).

$$ {}^{t}\mathbf{T}_{CoA}(\boldsymbol{\theta}^t)=\begin{bmatrix}{}^{t}\mathbf{R}_{CoA}(\boldsymbol{\theta}^t) & {}^{t}\mathbf{p}_{CoA}\\ \mathbf{0}_{1\times3} & 0\end{bmatrix} \tag{40}$$

In equation (40), the position translation vector ${}^{t}\mathbf{p}_{CoA}$ is determined by H_t (constant). The rotation matrix ${}^{t}\mathbf{R}_{CoA}(\boldsymbol{\theta}^t)$ is obtained by multiplying the three rotation matrices of the trunk, such as rolling, pitching and swinging $\boldsymbol{\theta}^t=[\varphi^t,\theta^t,\psi^t]^T$, and can be expressed as equation (41).

$$ {}^{t}\mathbf{R}_{CoA}(\boldsymbol{\theta}^t)=\mathbf{R}_z(\varphi^t)\mathbf{R}_y(\theta^t)\mathbf{R}_x(\psi^t) \tag{41}$$

In equation (39), ${}^{CoA}\mathbf{T}_0^k$ represents the reversible constant homogeneous transformation matrix from the left and right arm basic coordinate system $\sum_0^l$ to the double arm central coordinate system $\sum_{CoA}$, which can be expressed as equation (42).

$$ {}^{CoA}\mathbf{T}_0^k=\begin{bmatrix}\mathbf{1}_{3\times3} & {}^{CoA}\mathbf{p}_0^k\\ \mathbf{0}_{1\times3} & 1\end{bmatrix} \tag{42}$$

In equation (42), ${}^{CoA}\mathbf{p}_0^l=[0,0,-D/2]^T$ is the left arm position vector and ${}^{CoA}\mathbf{p}_0^r=[0,0,D/2]^T$ is the right arm position vector. In equation (39), ${}^{0}\mathbf{T}_7^k(\boldsymbol{\theta}^k)$ is the transformation matrix of the left (right) arm end effector coordinate system $\sum_7^k$ relative to the left (right) shoulder coordinate system $\sum_0^k$, which can be expressed as equation (43).

$$ {}^{0}\mathbf{T}_7^k(\boldsymbol{\theta}^k)=\prod_{i=1}^{7}{}^{i-1}\mathbf{T}_i^k(\theta_i^k) \tag{43}$$

In equation (43), ${}^{i-1}\mathbf{T}_i^k(\theta_i^k)$ is the homogeneous transformation matrix of the left and right brachial plexus $\sum_{i-1}^k$ to $\sum_i^k$ (i=1,2,…,7, $\mathrm{k}=l,r$ represents the left and right arms), which can be obtained by substituting the four **D-H** parameters （$\theta_i^k, \alpha_i^k, a_i^k, d_i^k$） of the connecting rod of the left and right arm 7 into equation (38).

Equation (39) ${}^{w}\mathbf{T}_7^k(\boldsymbol{\Theta}^k)=\begin{bmatrix} {}^{w}\mathbf{R}_7^k & {}^{w}\mathbf{p}_7^k \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}$ is the **DK** ($\boldsymbol{\Theta}^k$) equation in the form of **HRDA** homogeneous coordinate transformation. Expanding it can represent the **DK** ($\boldsymbol{\Theta}^k$) equation in vector form.

$$\mathbf{x}_e^k=\mathbf{f}^k(\boldsymbol{\Theta}^k) \tag{44}$$

In equation (44), $\mathbf{x}_e^k$ is the expression of pose vector $[x_e^k, y_e^k, z_e^k, \phi_e^k, \theta_e^k, \psi_e^k]^T$ of **HRDA** end pose matrix $\begin{bmatrix} {}^{w}\mathbf{R}_7^k & {}^{w}\mathbf{p}_7^k \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}$, $\boldsymbol{\Theta}^k=\left[[\boldsymbol{\theta}^t]^T,\ [\boldsymbol{\theta}^k]^T\right]^T$.Equation (44) means that the 6-dimensional $\mathbf{x}_e^k$ vector of the end pose of the left (right) arm of the humanoid robot is a function or mapping relationship between the 3-dimensional configuration $\boldsymbol{\theta}^t$ of the trunk and the 7-dimensional configuration $\boldsymbol{\theta}^l$ （$\boldsymbol{\theta}^r$） of the left (right) arm.

*5.3.2 Establishing **HRDA** inverse kinematics model*

HRDA inverse kinematics (**IK**) describes the geometric parameters of **HRDA** and trunk linkage mechanisms and the pose of double arm end effector relative to a specific reference coordinate system, judges whether **HRDA** end effector can reach the pose, and solves the rotation angle of each joint configuration to reach the desired pose. By deriving the time t on both sides of equation (44), the differential relationship (45) between **HRDA** configuration $\boldsymbol{\Theta}^k$ and posture $\mathbf{x}_e^k$ can be obtained.

$$\dot{\mathbf{x}}_e^k=\mathbf{J}_k(\boldsymbol{\Theta}^k)\dot{\boldsymbol{\Theta}}^k \tag{45}$$

In equation (45), $\dot{\mathbf{x}}_e^k$ is the generalized velocity of the end of **HRDA** in the workspace. $\dot{\boldsymbol{\Theta}}^k$ is the joint velocity. $\mathbf{J}_k(\boldsymbol{\Theta}^k)$ is the partial derivative matrix of 6x10, that is, the Jacobian matrix of **HRDA**. The elements in row i and column j ($\mathrm{k}=l,r$ represents left and right arms) can be expressed as equation (46).

$$J_{ij}^k(\boldsymbol{\Theta}^k)=\frac{\partial f_i^k(\boldsymbol{\Theta}^k)}{\partial\Theta_j^k}\ (\mathrm{i}=1,\ldots,6,\ \mathrm{j}=1,\ \ldots,10) \tag{46}$$

Obviously, there is redundancy in **HRDA** kinematics, and singularity problems will be encountered when solving **IK**. If the generalized inverse $\mathbf{J}_k^+(\boldsymbol{\Theta}^k)$ of $\mathbf{J}_k(\boldsymbol{\Theta}^k)$ exists, the **IK** differential expression (47) of **HRDA** can be obtained from equation (45).

$$\dot{\boldsymbol{\Theta}}^k = \mathbf{J}_k^+(\boldsymbol{\Theta}^k)\dot{\mathbf{x}}_e^k \tag{47}$$

For the convenience of numerical calculation, the incremental form of equation (47) can be expressed as (48).

$$\Delta\boldsymbol{\Theta}^k = \mathbf{J}_k^+(\boldsymbol{\Theta}^k)\Delta\mathbf{x}_e^k \tag{48}$$

In order to improve the operability and stability of **HRDA** control, this paper uses the **DLS-SVD** method integrating Damped Leased Square (**DLS**) and Singular Value Decomposition (**SVD**) technology to solve the numerical solution of **IK** (48) formula of **HRDA**.

5.3.3 HRDA software framework

The **HRDA** software framework designed in this paper is shown in Figure 1. Based on this framework, visual processing and motion estimation are processed in parallel and interdependent at the same time, which not only reduces the complexity of the algorithm, but also improves the utilization of resources, especially suitable for the computing processing of dual core CPU. The validity of the proposed algorithm is verified by the constructed **HRDA** experimental platform.

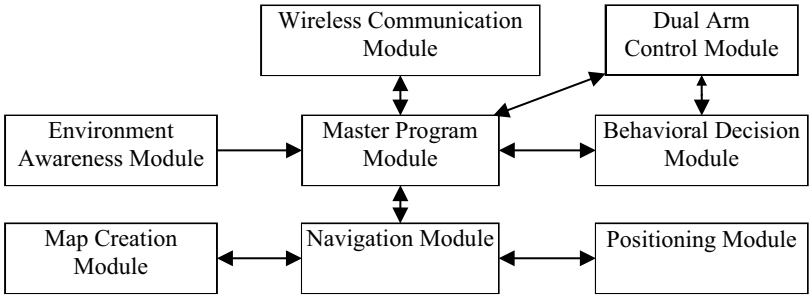

Fig. 1. HRDA software framework.

6 Conclusion

Based on the analysis of feature region location and information description in the region, this paper proposes a feature extraction algorithm based on **LPP**, which reduces the computational complexity of **SLAM** and the sensitivity to wrong data association. Aiming at the problems of phase winding and phase singularity in visual phase matching, a phase matching algorithm based on adaptive Gabor filter has been proposed. Using **UT** and selective acquisition strategy, **UTFastSLAM** algorithm based on **Rao-Blackwellised** has been proposed, which effectively reduces the particle dilution phenomenon of **PF** and significantly improves the calculation accuracy and robustness of **FastSLAM** algorithm. Finally, the **HRDA** experimental platform has been constructed, and the forward and reverse kinematics models of **HRDA** have been established to verify the effectiveness of the algorithm proposed.

The researches have been sponsored by the following scientific projects: Shenzhen basic research project (JCYJ20180307124010740), school level science and technology project (SZIIT2020KJ016, LHPY-2020007,LHPY-2020008), the seventh batch of school level education and teaching reform research and practice projects (10600-20-010201-06011).

References

1. A.Davision, I.Reid,N.Molton, and O.Stasse, MonoSLAM :Real Time Single Camera SLAM, IEEE Trans. Pattern Anal. Mach. Intell., 29(6):1052-1067,2007
2. Mark Cummins and Paul Newman, Accelerated Appearance-Only SLAM ,2008 IEEE International Conference on Robotics and Automation Pasadena, CA, USA, 1828-1883, 2008
3. J.Neira, A.J.Davision, J.J.Leonard, Guest Editoral Special Issue on Visual SLAM . IEEE Transactions on Robotics, 24(5):929-931,2008
4. A.J.Davision, Real-Time Simultaneous Localization and Mapping with a single camera, in Proc. Int. Conf. Comput. Vis.,Nice,France,2:1043-1410,2003
5. P.Smith, I.Reid, and A.Davision, Real-time Monocular SLAM with straight lines, in Proc.Brit. Mach. Vision Conf, 17-26, 2006
6. Michael J. Milford,Gordon F. Wyeth, Single Camera Vision-Only SLAM on a Suburban Road Network, 2008 IEEE International Conference On Robotics and Automation, CA, USA, 3684-3689,2008
7. J.Civera, A.J.Davision, J.Montiel, Inverse Depth Parametrization for Monocular SLAM IEEE Transactions on Robotics, 24(5):932-945,2008
8. Dorko G., Schmid. C. Selection of Scale Invariant Neighbourhoods for Object Class Recognition. In Proceedings International Conference on Computer Vision,634-640,2003
9. Goedeme T., Tuytelaars T. Van Gool L. Fast Wide Baseline Matching for Visual Navigation. In Proceedings IEEE Conference on Computer Vision and Pattern Recognition, 24-29,2004
10. David G. Lowe. Distinctive Image Features from Scale-Invariant Key points. International Journal of Computer Vision,60(2),99-110,2004
11. Ye K,Sukthankar R. PCA-SIFT: A more distinctive representation for local image descriptions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2004.1:551-517
12. R.Merwe, A.Doucet, N. Freitas, el al. The Unscented Particle Filter. Technical Report CUED/FINFENG/TR 380, Cambridge University, Engineering Department, 2000
13. M.Montemerlo and S.Thrun, Simultaneous Localization and mapping with unknown data association using FASTSLAM. Proc. IEEE Int.Conf. ICRA'2003, Taipei,2003
14. M.Montemerlo ,S.Thrun,D. Koller, et al. FastSLAM 2.0: A Novel Partical Filtering Algorithm for Simultaneous Localization and Mapping that Provably Converges. Proc. Of the Int. Joint Conference on Artificial Intelligence, 2003:1151-1156