A New Fast Histogram Matching Algorithm for Laser Scan Data

Quan Qiu
State Key Laboratory of Robotics
Shenyang Institute of Automation, Chinese Academy of Sciences
Shenyang, China
qiuquan@sia.cn

Jianda Han
State Key Laboratory of Robotic
Shenyang Institute of Automation, Chinese Academy of Sciences
Shenyang, China
jdhan@sia.cn

Abstract—**This paper presents a new fast histogram matching algorithm for tracking the position and orientation of robots without the help of any odometers. Histogram avoids extracting geometrical primitives from the sensor data, acts independently from odometers, and is easy to be implemented. Here, the authors employ it to analyze laser scan data. A new concept named "hierarchical histogram structure" is introduced for constructing and matching histograms. This structure makes full use of the information in a laser scan and speeds up the histogram matching to a satisfying level. Computational complexity analysis and experiments show the feasibility of the new algorithm both mathematically and factually.**

Keywords—**fast, histogram, matching, laser scan**

I. Introduction

Self-localization is an important research field in robotics. Usually, self-localization is the base of map building and self-guidance. A fast and precise localization helps robots to act smartly and wisely. Researchers have proposed many approaches for self-localization. Some of them use the data from inertial navigation sensors, such as odometers, inclinometers and compasses, to localize robots [1-2]. This kind of methods is called dead reckoning. Unfortunately dead reckoning often leads to accumulation of drift errors. Some researchers localize their robots using landmarks [3]. This strategy needs prior information about the environment, so it is not available in unknown environments. Others present algorithms which can achieve the goal of self-localization without odometers [4]. Map matching is one of these algorithms. By comparing two local maps which are constructed with the sensor data sampled at two adjacent places, map matching estimates the orientation and translation differences between them. If we know the initial position, we can localize all the sample points by adding up the orientation and translation differences. This strategy is available in unknown environments and works independently from odometers, so it plays a more and more important role in self-localization.

Map matching has several variants for matching laser scans. These variants use different mathematical tools and act diversely. Cox [5] forms a cost function using the distances from points to segments, and computes the orientation and translation shifts when the cost function is minimum. The segments are extracted from a preconcerted reference scan. Each point in the other scan is assigned to a target, and the target is the nearest segment in the reference scan. Then the cost function is the sum of all the distances between the points and their targets. The Cox algorithm is easy to be understood, but it has to extract geometrical primitives from the sensor data and has to compute inverse matrices. So Cox doesn't fit unstructured environments and has much difficulty in operating maps with a large quantity of points. Other than Cox, IDC [6] introduces correspondence pairs to overcome the embarrassment in unstructured environments. It chooses two strategies to find the correspondence points, and uses iteration to search for the best orientation and translation estimations. In the two strategies, one is good at getting orientation values, and the other is accomplished in determining translation values. IDC takes the long points of them and fuses the two strategies into a perfect "dual" algorithm. Iteration guarantees the precision of IDC algorithm, but slows down the calculation process. Also, finding the correspondence pairs is a bothersome task. Histogram is another good choice for unstructured cases. Instead of extracting geometrical primitives from the raw data, this approach stores the geometrical information in the angle and translation histograms. Correlation, not cost functions, is the main mathematical tool to carry out the matching step. Constructing histograms is much easier than finding correspondence points, and the computational complexity is irrelative to the number of scan points. So compared with Cox and IDC, histogram has innate excellent qualities.

Histogram was first used to match laser scans in 1994 [7]. After that, this method has been widely used in navigation, localization and mapping [8-9]. Recently, Michael Bosse and Jonathan Robert developed a histogram approach to recognize a previously mapped area in a large global map [10]. The histogram algorithm seems mature enough and all we need to do is just to use it in different cases. The authors don't think so. Through some smart improvement, we can make histogram acquire an even better performance. In this paper, hierarchical histogram structure is employed to match the histograms in a more efficient way. This brand new structure reforms the histograms to pay more attention on the interested sections, and simplifies the traversing matching process to a small number of correlation operations. As a result, the calculation speed of histogram algorithm is enhanced greatly and the new histogram algorithm becomes more qualified in real-time applications.

This article is organized as: Section 2 explains the construction of angle histogram and translation histogram for laser scans; Section 3 introduces hierarchical histogram structure and makes a rough comparison between the new structure and the original one; Section 4 demonstrates the feasibility of the new histogram algorithm with experiments; Section 5 sums up the new algorithm and gives a draft plan of the future work.

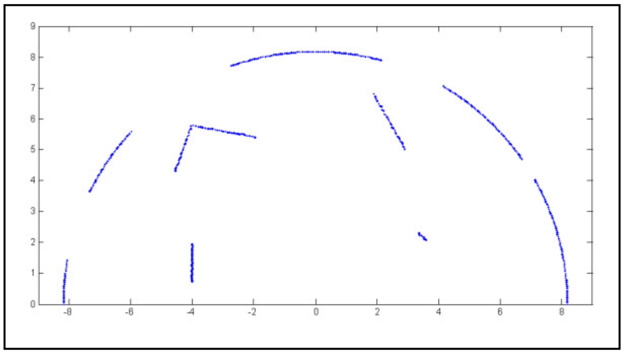

Figure 1. A single laser scan

II. CONSTRUCTION OF HISTOGRAMS

Histogram gives an easy and fast solution for orientation and position estimation. Other than geometrical primitive algorithms, histograms don't extract geometrical primitives from the laser scan data, but combines the geometrical information into the histograms. Then making good use of the connotative clues to construct histograms for accurate orientation and translation estimation becomes very important. To illuminate the histogram construction strategy clearly, the authors will divide this section into two parts: angle histogram construction and translation histogram construction.

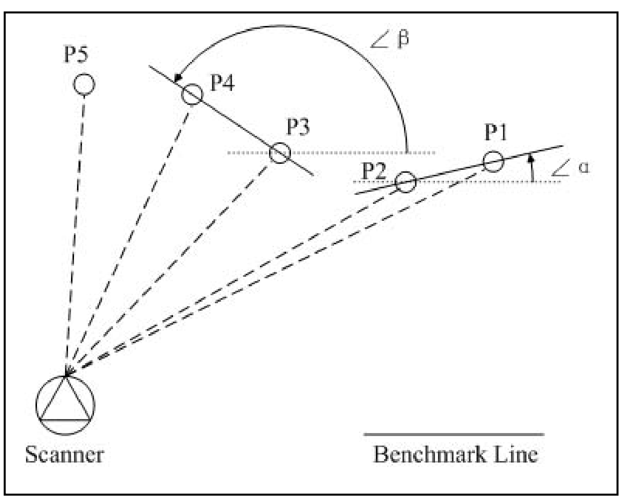

Figure 2. Adjacent vector angle

A. *Angle Histogram*

A laser scan is a serial of numbers which stand for the distances from the scanner to the nearest obstacles in given orientations. In general, the distances queue according to their corresponding orientation angles in an average ascending order. Figure 1 shows a single scan obtained by SICK LMS-221 laser scanner, and this laser scanner can give a scan of 180 degrees.

The process of constructing an angle histogram is the process of extracting geometrical information from the scan data. Finding the angles and filling them into the angle histogram is the main task in this step. An angle histogram is divided into 360 intervals. Each interval stands for a degree and is a counter of the angles which lie in it. We can call each angle an "adjacent vector angle", because it is cornered by a benchmark vector defined beforehand and a vector which links two adjacent scan points. We can understand the process of finding adjacent vector angles well with the help of Figure 2. The segment dash lines stand for the radials coming from the scanner. Pi denotes a scan point, which probably means an obstacle point. The segment that links P3 and P4 is an adjacent vector, the point dash line is the benchmark vector, and naturally, angle β is an adjacent vector angle. In the same way, we know α is an adjacent vector, too. Now, computing the values of adjacent vector angles remains nothing but simple vector operations. After getting the value of an adjacent vector angle, we can insert it into the angle histogram easily. For example, if we have an adjacent vector angle of 30 degrees, we insert it in by pulsing 1 to the counter of the corresponding histogram interval that contains the angle of 30 degrees. An example of angle histogram can be found in Figure 3.

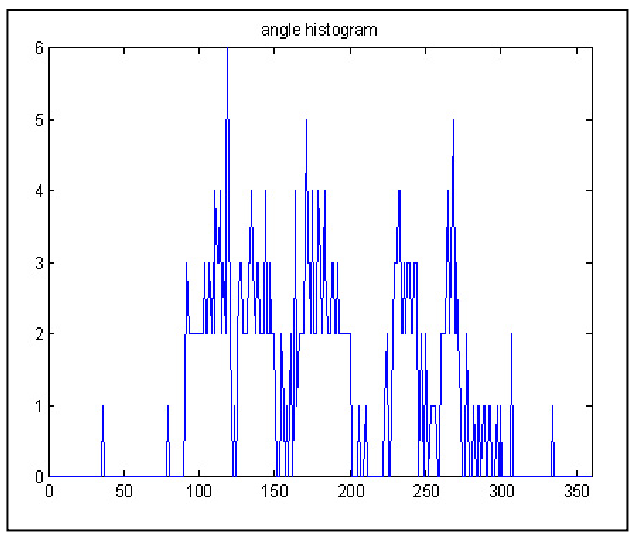

Figure 3. Angle histogram

B. *Translation Histogram*

After using the angle estimation to correct the angle drift between two sets of laser scan data, there comes the next phase of histogram estimation: translation estimation. Similar to the angle estimation, translation histograms need to be constructed first. In fact, translation histogram is an honest display of the distribution for measurement points in x- or y- direction. It is formed by a number of intervals which is mainly determined by the maximum error of the measurement. The number of intervals can be obtained using the full range and the interval size. While the angle histogram has a changeless interval number of 360 and a circular sequence, translation histogram has an alterable number of intervals based on the features of the laser scanner and a linear sequence. Meanwhile, translation histogram not only has positive part but also has minus part, just like a coordination axis. These discrepancies make the construction process of translation histogram a little different from that of angle histogram. When we construct a translation histogram, we first determine the number of intervals and the positions of the positive part and minus part. Then we project the scan point into a coordination to get its values in x- or y-direction. To insert a measurement point into a translation histogram, we should find the interval incepting the point first. The equation used to compute the interval index is

$$Index(X_i) = abs(X_i) \setminus size \tag{1}$$

where "\" is the modulo operator. If X_i is positive, we insert the point into the positive part of the translation histogram, otherwise, we insert it into the minus part. Figure 4 gives us an example of translation histograms. The two histograms are constructed using the same scan data as Figure 1. Here, we let the interval 0.02 meters long, and each histogram has more than 800 intervals when the full range is 8.183 meters. Both of the two histograms have minus parts in the left and positive parts in the right. We can see the minus part of the y histogram is empty, just because the scan set has not minus components.

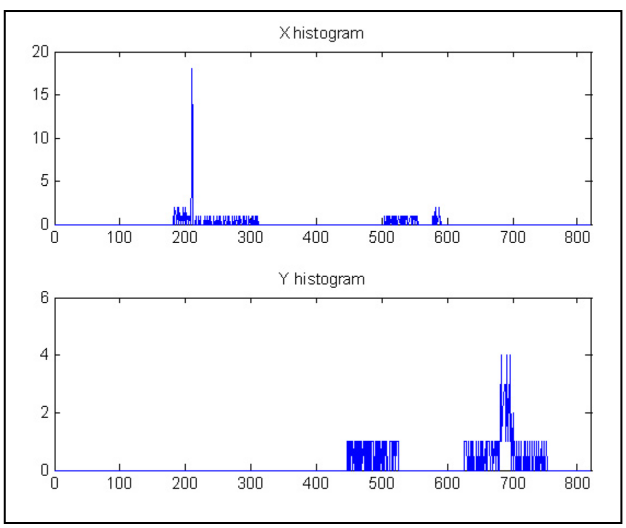

Figure 4. Translation histograms

III. MATCHING HISTOGRAMS

A. Crosscorrelation

Crosscorrelation is the basic mathematic tool for the matching step. The definition function of crosscorrelation is given through

$$C(y) = \lim_{x \to \infty} \frac{1}{2x} \int_{-x}^{x} f(x) g(x+y) dx \,. \tag{2}$$

In this function, the integration result $C(y)$ is a measurement for the correlation between two stochastic functions, $f(x)$ and $g(x)$, regarding the phase-shift y. $C(y)$ will have an absolute maximum at s, when $f(x)$ equals $g(x+s)$. Based on this point, we can estimate the drift between two histograms by searching for the phase shift leading to the maximum value of the crosscorrelation function. (2) is the expression for continuous cases. Histograms of laser scan data are discrete, so the crosscorrelation formula needs to be transformed into a discrete form. The new formula could be written as

$$C(j) = \sum_{i=1}^{n} h_1(i) \bullet h_2(i+j) \tag{3}$$

in which h_1 and h_2 are the histograms of two laser scans, n is the number of intervals in a histogram. (3) is the most useful function in histogram matching.

B. Hierarchical Histogram Structure

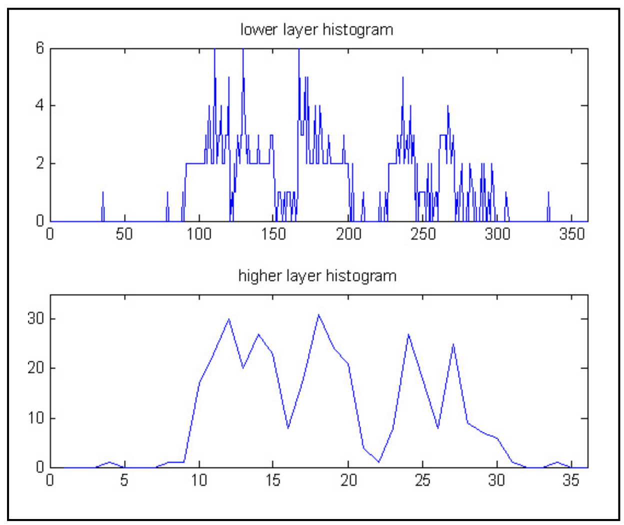

Figure 5. Hierarchical histogram structure

Histogram matching is a process of finding the proper orientation and translation warps between two histograms. The warps will generate a translation matrix. If the matrix can correct one scan to make it have almost the same position and orientation as the other one, we say that the match is successful. Generally, we use traversing method to search for the warps. That is, if the histograms to be matched both have n intervals, the traversing method needs to compute at least n crosscorrelations and then finds the biggest value among them. As we all know, traversing method is a good method counting in all the information of the histograms. But sometimes, our instinct tells us that we don't have to act in such a complicated way. Most histograms have uneven distribution among their intervals, and we can usually find out where the most important parts are and then conclude the approximate drift between two similar histograms easily. There must be other approaches which can find the maximum crosscorrelation more effectively than the traversing method.

The authors present a new method using hierarchical histogram structure to fulfill the searching task. Hierarchical histogram structure means that there are at least two pairs of histograms belonging to different layers to do the matching action. Different layers stand for different interval resolutions. Taking the angle histogram for an example, a higher layer may have an interval resolution of 10 degrees, and a lower layer may have an interval resolution of 1 degree. Then the matching step could be divided into two parts. First, we match the pair of histograms in the higher layer to find the most interested subarea, usually a big interval containing several small intervals. Second, we concentrate on the allocated subarea, traverse in it, and search for the maximum crosscorrelation of histograms in the lower layer. In this way, we can get rid of the meaningless computations for the unimportant subareas. As a result, the matching step is speeded up greatly. Figure 5 shows a two-layer hierarchical structure. We can see that the interested subareas become distinguish in the higher-layer histogram.

C. *Computational Complexity Analysis*

There is no doubt that the new matching approach using hierarchical histogram structure is faster than the ordinary traversing approach. But a simple presentation is not reasonable enough. A rigorous mathematic demonstration obviously speaks louder. Now we prove the validity of the hierarchical histogram structure through computational complexity analysis.

For convenience, we use the original histograms as the lower layer ones. Higher-layer histograms are constructed based on the lower-layer ones, and we call the new constructed histograms rough histograms. Assuming the original histograms both have n small intervals, the rough histograms both have N subareas(big intervals), we know that each subarea of the rough histogram has n/N small intervals. Let's consider the ordinary traversing method. To find the biggest crosscorrelation value, we need to get all the crosscorrelation values, so the computation of crosscorrelation needs to be repeated for n times. When calculating a crosscorrelation value, we have to do n times of multiplication and n times of addition. Then we can infer the computational complexity of the ordinary traversing method is

$$O_{ordinary} = 2n^2 . \quad (4)$$

Because the rough histogram has N subareas, using the result of ordinary traversing method, we can know the computational complexity of finding the interested subarea is $O(2N^2)$. To determine the final maximum crosscorrelation, n/N times of crosscorrelation calculation are inevitable, and that takes $2n^2/N$ times of operation. Summing up all the operations, we have the total computational complexity of the new approach as

$$O_{new} = 2N^2 + 2n^2/N . \quad (5)$$

We can demonstrate the decrease in calculation with an instance angle histogram. An angle histogram usually has 360 small intervals. Let the rough histogram has 36 subareas. In this example, n is 360 and N equals 36. Substituting n into (4) and N into (5), we obtain that $O_{ordinary}$ equals 259200 and O_{new} equals 9792. The hierarchical histogram structure method cuts down 96% unimportant operations from the traversing approach. That's a great progress. In practice, the improvement differs according to the environment conditions. The authors will give some experiment results to show the availability of the new approach.

IV. Experiments

A. *Computational Facility*

In this paper, the authors dedicate most attention to simplifying the matching process and saving the computational time. Here, we prove the convenience of the new algorithm through some experiments. We use LMS 221 laser scanner as the main sensor to collect scan data. All programs of the experiments are coded in the MATLAB7.0 environment. The computer has an AMD Athlon(tm) 64 processor with a working frequency of 2.20GHz.

First, an experiment about the angle match is carried out. In this experiment, we use the new algorithm and the traditional algorithm to match 1000 pairs of scans respectively, repeat the matching for 5 times and record the running durations in Table I. The higher layer histogram has 36 subareas, and each subarea contains 10 small intervals of the lower layer histogram. We can find from the table that the hierarchical structure strategy saves about 40 percent of the computational time than the traversing strategy.

TABLE I. Angle Match Durations

Experiment numbers	**1**	**2**	**3**	**4**	**5**
Traditional algorithm	8.2500s	8.2656s	8.3907s	8.3906s	8.4062s
New algorithm	4.6406s	4.6562s	4.7657s	4.7813s	4.7812s

Then, we see how fast the new algorithm acts in translation matching. The laser's full range is 8.183 meters. We let the small interval of the lower layer histogram be 0.02 meters, and the subarea of the higher layer histogram contains 25 small intervals. We match 1000 pairs of scans with two algorithms respectively for 5 times, and the durations are listed in Table II. Table II shows the new algorithm decreases the translation match time greatly. The new algorithm is about 20 times faster than the traditional algorithm. The hierarchical histogram structure plays its magic in translation match more obviously than in angle match. The reason for this phenomenon is that the construction step of the angle histogram occupies most of the computational time and makes the influence coming from the new algorithm not as apparent as that in translation match.

TABLE II. TRANSLATION MATCH DURATIONS

Experiment numbers	1	2	3	4	5
Traditional algorithm	55.6250s	55.6562s	55.6719s	55.7619s	55.6406s
New algorithm	2.9687s	2.7969s	2.8125s	2.7657s	2.7870s

Finally, we put the angle match and the translation match together and the experiment is also repeated 5 times respectively. The whole durations of the two algorithms are shown in Table III. The new algorithm cuts off four fifths running time from the traditional algorithm. We can easily conclude that the new algorithm equipped with hierarchical histogram structure is more suitable for real-time applications than the traditional algorithm.

TABLE III. THE WHOLE MATCH DURATIONS

Experiment numbers	1	2	3	4	5
Traditional algorithm	68.8906s	68.8438s	68.8750s	68.8594s	68.8750s
New algorithm	13.7188s	13.6875s	13.6875s	13.7656s	13.6719s

B. Matching Accuracy

The hierarchical structure strategy improves the running speed of the histogram algorithm so greatly that we can't help doubting that if the new algorithm makes some compromises between the matching precision and the matching speed. In fact, from the mechanism of the new algorithm, we know there are no compromises. That is, the new algorithm speeds up the matching step without sacrificing the matching accuracy. To demonstrate this point, the authors give another experiment. In the experiment, we let the laser scanner scan the same scene for thousands of times continuously. Then we select 1000 pairs of scans from the whole data set, and these scans differ from each other because of the random errors caused by the scanner. The new algorithm and the traditional algorithm are employed to match the 1000 pairs of scans and the occurrences of the errors are recorded in Table IV. Analyzing the table contents, we find the new algorithm gives the same experiment result as the traditional one. Furthermore, all the errors appear at the same matching pairs. This experiment strongly proves that the hierarchical histogram structure strategy makes no change in the matching accuracy, and the new algorithm always runs with the same precision as the traditional one.

TABLE IV. MATCHING ERRORS

Error types	Angle match		Translation match		
	$\pm 1^\circ$	$> \pm 1^\circ$	$\pm 0.02m$	$\pm 0.04m$	$> \pm 0.04m$
Traditional algorithm	15	2	6	9	2
New algorithm	15	2	6	9	2

V. CONCLUSION AND FUTURE WORK

We present a new fast histogram algorithm for matching laser scans. In the new algorithm, hierarchical histogram structure is employed to improve the calculation efficiency of the histogram algorithm. A higher layer histogram takes charge in finding the most interested subarea in the lower layer histogram, and usually the phase shift that makes the maximum crosscorrelation is contained in the most interested subarea. In this way, the new algorithm avoids of traversing through the whole histogram to find the wanted phase shift and saves much computational time. In fact, this strategy is universal in all histogram matching applications and can speed up the matching step quite effectively. Some experiments are done to make a comprehensive analysis of the new algorithm. From the analysis, we know that the new algorithm can really decrease the computational consumption greatly, and at the same time, it can keep a match accuracy as same as the traditional one.

Because the match precision is not satisfying enough, further study will focus on enhancing the accuracy of the histogram matching algorithm. Some new histogram construction strategy may be introduced, and a mechanism of filtering the wrong matching results will also be developed.

ACKNOWLEDGMENT

The authors would like to thank Jinghong Jin, Xingang Zhao, and Chen Zhong for the heuristic discussion, implement assistance, and kind encouragement.

REFERENCES

[1] J. S. Gutmann and C. Schlegel, "Amos: Comparison of scan matching approaches for self-localization in indoor environments," proceedings of the 1st Euromicro Workshop on Advanced Mobile Robots, IEEE Computer Society Press, 1996.

[2] H. Afaska, M. Zribi, and M. Benjelloun, "Information fusion sensors for vehicle localization using evidence theory," IEEE International Conference on Industrial Technology, 2004.

[3] S. Thrun, "Bayesian landmark learning for mobile robot localization," Machine Learning, vol. 33, No. 1, 1998.

[4] F. Amigoni, S. Gasparini, and M. Gini, "Map building without odometry information," proceedings of the 2004 IEEE International Conference on Robotics & Automation, New Orleana, LA, April 2004.

[5] I. J. Cox, "Blanche—An experiment in Guidance and navigation of an autonomous robot vehicle," IEEE Transactions on Robotics and Automation, vol. 7, No. 2, April 1991.

[6] F. Lu and E. Milios, "Robot pose estimation in unknown environments by matching 2D range scans," Journal of Intelligent and Robotic Systems, vol. 18, 1997, pp. 249-275.

[7] G. Weiβ, C. Wetzler, and E. v. Puttkamer, "Keeping track of position and orientation of moving indoor systems by correlation of range-finder scans," IEEE/RSJ International Conference on Intelligent Robots and Systems, Munich, Germany, September 12-16, 1994.

[8] T. Röfer, "Using histogram correlation to create consistent laser scan maps," proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems, Lausanne, Switzerland, October 2002.

[9] N. Kateyas, S. Voliotis, T. Zahariadis, and K. Hrissagis, "Range histogram: an environment representation method for in-door navigation of mobile robots," International Symposium Electronics in Marine, Zadar, Croatia, June 16-18, 2004.

[10] M. Bosse and J. Roberts, "Histogram matching and global initialization for laser-only SLAM in large unstructured environments," proceedings of IEEE International Conference on Robotics and Automation, Roma, Italy, April 10-14, 2007.