Mosaic Based Flexible Navigation for AGVs

André Lucas, Camilo Christo, Miguel Pedro Silva, Carlos Cardeira
IDMEC/IST TU-Lisbon, Portugal
{cchristo@dem., mps@,carlos.cardeira@}ist.utl.pt

***Abstract*-Highly flexible systems require that Automatic Guided Vehicles (AGVs) in a plant navigate autonomously and changes in their missions should not require difficult setup procedures. In this paper we address the problem of localization of a mobile robot in a indoor environment. The robot is able to find its position without any grounded wires, landmarks or laser beacons. The robot uses images acquired in the roof to compute its position and navigate between coordinates. The main contribution is the absence of external services to solve the AGV localization problem, allowing fast reconfiguration.**

I. INTRODUCTION

Today a manufacturing plant has to be flexible and seamless reconfigurable. Currently, a large amount of the total cost of a manufacturing plant over its lifetime is spent on installation, setup and reconfiguration. If a plant is subject to changes in its process flow or changes due to the introduction of new or replacement of noncompetitive equipment that is provided by different makers, then the downtime and lifetime costs rise considerably [James 2005].

The problem of the inflexible communication infrastructure can be largely eased by the use of wireless technologies [Cardeira 2006]. The problem of software reconfiguration is being addressed by several ongoing works namely by the use of service oriented architectures [Bepperling 2006].

In this paper we address the localization problem and we present a highly flexible localization system for Automatic Guided Vehicles (AGVs), as a way to ease the problem of locating a robot in a dynamic environment.

The localization system is highly reconfigurable as the robot builds a map of the environment and uses it to define its localization in the environment. At any time, the robot can rebuild the map to self reconfigure to environment changes.

A. Guiding AGVs

There are several different methods to guide AGVs. Wire-guidance is the simplest form of navigation. An RF signal is transmitted from a wire buried in a slot below the floor to a sensor under the vehicle. The sensor detects the signal and adjusts the position of the vehicle to keep it on the path. Because the slot must be cut into the floor, wire-guided systems are most commonly used where paths are unlikely to change [Trebilcock 2007].

However, when fast reconfiguration is a must, more autonomous and flexible ways to let the AGVs navigate are necessary. Some solutions rely on the use of beacons along the factory that act like GPS for triangulation. However if we want the AGV to be really autonomous, we should not rely on external systems. It is up to the AGV to autonomously build a map and guide itself in the environment. At least, the AGV should be guided through the previously define track and then be able to reproduce that track without accumulating errors like a teach pendant robot.

B. MBLAM - Mosaic-based Localization and mapping

The simultaneous localization and mapping is nowadays an area of big interest and under great investigation, it combines methodologies of several areas like pose-estimation, computer vision, matrix algebra, etc. In this work is made and tested a different approach that presents a solution to the problems associated with the localization and map building. The developed methodology consists in a Mosaic-based Localization and Mapping (MBLAM). The methods used are focused in missions where the vehicle is ordered to navigate in an unknown environment and build a map of the environment while locating itself in this map. Using MBLAM the robot builds maps from video visual mosaics. As far as new image frames overlap the precedent image, the robot continues building the overall map. When the map is built, the robot compares the actual image to the map to find where the image is in the global map. Apart from calibration procedures, the localization of the image in the map has a direct correspondence with the localization of the robot in the environment. The robot is hence able to determine its position by a comparison of the actual view with the global view of the map previously built. The "map" is indeed the global image of the environment and as long as the robot can find the actual image in the global image, it is actually find its position in the map.

C. SLAM - Simultaneous Localization and mapping

Simultaneous Localization and Mapping (SLAM) is perhaps the central information engineering problem in mobile robotics research. Being simple to state but challenging to solve: The Simultaneous Localisation and Mapping (SLAM) problem asks if it is possible for a mobile robot to be placed at an unknown location in an unknown environment and for the robot to incrementally build a consistent map of this environment while simultaneously determining its location within this map. A solution to the SLAM problem has been seen as a `holy grail' for the mobile robotics community as it would provide the means to make a robot truly autonomous." [Durrant-Whyte, 2006]. The main difficulty of the localization and mapping becomes from the interaction between the robot and the environment where it operates, normally associated with noisy sensing and measuring.

Solutions to the issues around simultaneous localization and mapping are very important due to their contribution to increase the autonomy of mobile robots.

There is a need for methods that allow a larger autonomy in real world navigation attending to the fact that some other position systems might be inefficient or simply unavailable (e.g GPS). Nowadays there is a big incentive to the use of localization and mapping methodologies in new fields as well as a growing commercial interest in new applications like low cost cleaning robots for domestic usage or even touristic guides operating in like museums. Without this approach it would be necessary the use of expensive devices (like predefined tracks) which are much less flexible and require changes to the environment.

D. *RELATED* WORK

Mechanisms and methodologies have been developed for mobile robots in order to build a map of an unknown environment and, simultaneously, use it to navigate. There is a wide field of applications, from remote operate robots that work in other planets like Mars to their usage in unmanned robots working in the bottom of the ocean. Some of these methodologies are based on the same principle, the use of an estimate of the mobile robot pose based on landmarks. Among these approaches the most common are those that apply the Extended Kalman Filter (EKF), the Sparse Extended Information Filter (SEIF), the Unscented Kalman Filter (UKF) and the particle filters (FastSLAM).

One implementation well succeeded of the EKF was accomplished by [Davison and Murray, 2002], using an active vision scheme in a vehicle operating in a small environment that was rich in easily extractable landmarks. The vehicle was able to execute trajectories and also accurately determine its position, with errors scaled to centimeters. Another solution, this one base on SEIF, was reached by [Liu and Thrun, 2003], to an outdoor environment using an automobile as a mobile platform and where the results were similar to the usage of the EKF but with improvements in the computational complexity and bigger ability to use high dimension maps.

In [Sunderhauf et al., 2007] was presented a solution to this problematic using the UKF with monocular vision and based on simulation results they support the applicability of this methodology to real world systems(in this case an aircraft)

Results of the implementation of a particle filter where reached by [Barfoot, 2005], with a stereo camera system and visual landmarks extracted by the method of Scale Invariant Feature Transform (SIFT). Pose estimations at 3Hz with an error of 4% to all the distance traveled were obtained.

In order to solve the problem of the simultaneous localization and mapping an approach based in visual mosaics was implemented both in land and subsea applications.

[Blanc et al. 2005] used the ceiling as reference and was able to build a map of visual mosaics, navigate and determine its position.

In subsea applications, the method of mosaic based navigation was applied to an unmanned vehicle while it operated in the bottom of the ocean in order to determine an accurate estimate of the robot’s position [Gracias et al., 2003].

Several approaches are currently being developed for the simultaneously localization and mapping, mostly with the purpose of increasing the autonomy of robotic systems.

II. MOSAIC-BASED LOCALIZATION AND MAPPING

The present work is based on the methodology of using visual mosaics to simultaneously build a map and estimate the position. To accomplish that goal is necessary to overcome the three following steps: mapping, localization and navigation. The scheme of the mapping process is illustrated in the diagram below (Figure 1):

Those are the four steps that allow building a mosaic able to support the MBLAM methodology.

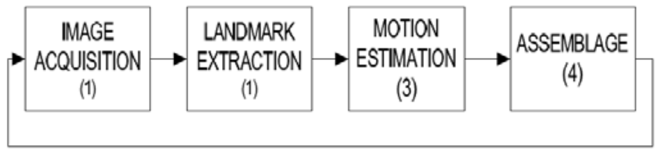

Fig. 1. Mosaic Construction.

A. *Image Acquisition*

A common practice to acquire images is the use of two cameras and implement a stereovision system able to make a 3D view of the environment. We did not follow this approach and we used just an ordinary web cam. The monocular system was chosen attending to:

• The increase of the computational burden associated with images captured by two different sources.

• Higher complexity when operating with stereo vision systems as well as the need of more accurate and expensive material in order to achieve a good performance.

During the development of the MBLAM it was assumed that the camera did not suffer from image border distortion.

B. *Landmark Extraction*

In this work, the Harris and Stephens corner detector [Harris and Stephens, 1988] was used to extract visual references. This method determines the matrix of the corners metric values in the images. This corner detector avoids the computational burden associated to other methods by alternatively calculate the following matrix of metric values:

$$R = AB - C^2 - s(A + B)^2$$

Where:

$$A = (I_x)^2 * w$$
$$B = (I_y)^2 * w$$
$$C = (I_x I_y)^2 * w$$

I_x and I_y are the gradients of the processed image in the correspondent directions and the operator $*$ is the convolution operator. The variable s is associated with the sensibility factor which is related to the probability of finding more corners. At last w is the matrix of the filtering coefficients, avoiding the detection of small and meaningless variations.

The result of these operations is a detector distinguishable by being robust to light changes and also big variations of the viewpoint.

Figures 2 and 3, show an image before and after the application of this corner detector. We draw the attention that not all detected corners are real corners. This could be achieved by a manual tuning of s, but we prefer to put the obtained image "as is" to illustrate the actual obtained result. Many corners are false corners but the important is that enough real corners are found. Real corners will likely be present in other views of the same scene. This will be important for the motion estimation.

Fig. 2. Building view

Fig. 3. Harris corner detection of the building view

C. *Motion Estimation*

The motion estimation method is the heart of the whole process of building the visual mosaic. It is responsible to find the outliers between the landmarks extracted in the images. It also determines the transformation matrix to apply to an image so it can find a best fit to the previous one with the maximum point correspondence between them. This is the step that allows the correct assemblage of the frames sequence and it is based on the *RANdom SAmple Consensus* (RANSAC) algorithm.

The RANSAC algorithm relies on a distance threshold. A pair of points, p_i^a (image a, set of points 1) and p_i^b (image b, set of points 2) is an inlier only when the distance between p_i^b and the projection of p_i^a based on the transformation matrix falls within the specified threshold.

The distance metric used in the RANSAC algorithm is obtained as follows:

$$d = \sum_{i=1}^{N} min\left(D\left(p_i^b, \varphi(p_i^a : H)\right), t\right)$$

Where:

p_i^a is a point in image a

p_i^b is a point in image b

$\varphi(p_i^a : H)$ is the projection of a point in image a based on transformation matrix H

$D(p_i^b, p_j^b)$ is the distance between two point pairs on image b

t is the threshold

N is the number of points

In a projective transformation the resulting image is achieved by the following equation applied to each one of the points (x, y) of the initial image:

$$[u_p, v, w_p] = [x, y, w]H$$

where H is a matrix 3-by-3 projection matrix.

Changing the coordinates according to the following equations:

$$u = \frac{u_p}{w_p}$$
$$v = \frac{v_p}{w_p}$$

resulting the next equivalent expressions:

$$u = \frac{h_1 x + h_4 y + h_7}{h_3 x + h_6 y + 1}$$

$$v = \frac{h_2 x + h_5 y + h_8}{h_3 x + h_6 y + 1}$$

where u and v are the coordinates of each point after the projective transformation.

For a better understanding of the result of such transformations figures 4 and 5 provides an example:

In the image of figure 4, after applying the following projective transformation matrix h, we obtain the image in figure 5.

$$h = \begin{bmatrix} 1 & 0.2 & 0.001 \\ 0.1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Fig. 4. Original image

Fig. 5. Image after the projective transformation

D. *Assemblage*

The principle of acquiring the visual mosaic is not complex, it consists simply in stitching the images captured in an instant k after applied the projective transformation H, to the global mosaic at instant $k - 1$. By repeating consecutively this process the complete mosaic is achieved.

Figures 6, 7, 8 and 9, present the creation of a mosaic using the capability of the process described by landmark extraction, motion estimation and final assemblage.

III. LOCALIZATION

The localization of the robot is made assuming the robot is in the center of the image. Actually, in indoor environments, we used a robot with a camera pointing to the ceiling. Attending to the fact that the robot navigates in the ground plan the position of the robot corresponds to the center of the actual frame (instant k) in the global map built iteratively. The orientation of the camera is similar to the *minerva* robot [Thrun et al., 1999], where the camera is also pointing to the ceiling in order to attain the best results.

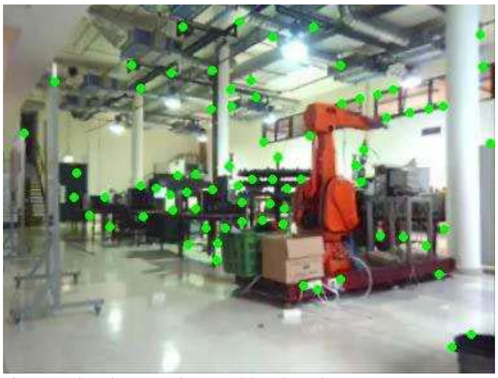

Fig. 6. First image: detected landmarks

Fig. 7. Second image: detected landmarks

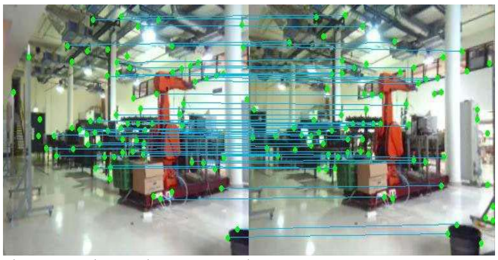

Fig. 8. Estimated correspondence

Fig. 9. Final assemblage

This way, the position of the robot is determined in the global map built according to the following image (figure 10):

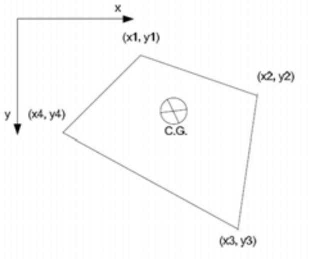

Fig. 10. Frame centroid

The determination of the centroid is obtained by:

$$X = \frac{\left(\frac{x_1 + x_2}{2}\right) + \left(\frac{x_3 + x_4}{2}\right)}{2}$$

$$Y = \frac{\left(\frac{y_1 + y_2}{2}\right) + \left(\frac{y_3 + y_4}{2}\right)}{2}$$

This way, the position of the robot is determined in the global map built.

IV. NAVIGATION

A. *Mobile platform*

The mobile robot Rasteirinho was the platform used in this work, it is a low cost robot built with the purpose to provide a better interaction between the theory knowledge acquired and their interesting practical applications [Cardeira and Sa da Costa, 2005]. The robot has an electrical circuit that emulates a serial communications port into USB and a 16F876A PIC. For this work, changes were made to the robot. Each wheel has now an associated encoder and a metal structure was built to support the PC(Personal Computer) and also the camera. The final assemblage of this device may be seen in picture 11.

Fig. 11. Robot "Rasteirinho"

In order to achieve the desired velocity in each wheel to navigate over predefined trajectories it was used a low level controller operating in the PIC 16F876A. This controller is responsible to make the bridge between the information ordered by the software on the PC and the velocity attained by the robot.

V. RESULTS

For the implementation of the developed MBLAM methodology we based our system in the Matlab demo Video Mosaicking available in the Video and Image Processing Toolbox. It was modified, improved and adjusted in order to create a video mosaic from a video sequence. This mosaic allows creating the map of the environment where the robot operates.

After the proper adjustments, the resulting model for video mosaicking allows to create panorama views like the one in figure 12 and also a map, that is it main purpose in this work. By integrating the process of map creation with the methods of localization and navigation, we achieved an approach that works as a solution to the problem of simultaneous localization and mapping, the Mosaic-Based Localization and mapping.

Fig. 12. Panorama view

Several experiments and tests were made to define the applicability of this approach to real-time application in an indoor environment. Here we present one experiment that refers to the ability of the MBLAM to provide the robot with the ability of navigate and control its trajectory based on its global position. The robot is able to follow a straight line in spite of the accumulated errors of the encoders that tend to make the robot deviate from the predefined trajectory.

B. *Controlled straight line*

For this test the trajectory was a straight line, the control methodology consisted in increasing or decreasing the speed of the robot wheels in order to stay in a previously defined gap. As soon as the robot crossed the limits of the gap the velocity of the proper wheel would be increased. Specifically, if it crossed the upper limit (410pixel), on the right, the speed of the right wheel would be increased by 1rad−1. The same would respectively happen if it crossed the lower bound. This way the robot followed a reference line for x = 400pixel.

Figure 13 shows the robot trajectory as seended from the camera pointed to the ceiling. The experiments presented above denote the successful usage of visual mosaics to create a map that is used to support the navigation, which is based on the global positioning. The whole process was performed simultaneously and in real-time.

In this work was also started a methodology that allows the robot to determine its position in a previously built map of the environment, by the usage of SIFT descriptors. In this method, a current frame of the environment is compared to the global map in order to determine the robot's location. This procedure is useful when the map is already built.

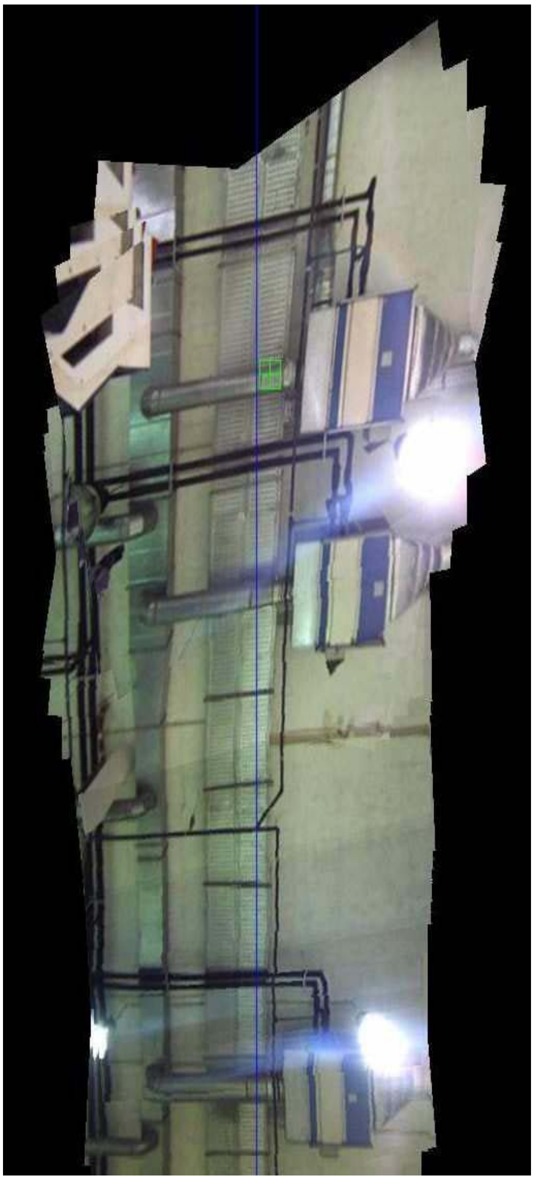

Fig. 13. Global map, controlled straight line

VI. CONCLUSION

This paper presents the methodology of Mosaic Based localization and mapping as a solution to the simultaneous localization and map building problematic. It provides the robot with the autonomy of simultaneously navigate, locate and build a map in an unknown environment. All the methods are exclusively vision-based and processed simultaneously at real-time. In the implementation was used a wheeled low cost robot operating in an indoor environment using the ceiling as visual reference to build a map through visual mosaics. The robot is able to drive around autonomously and create its world representation as well as execute predefined trajectories due to its control methods based on the global position.

ACKNOWLEDGMENT

This work was partly funded by the FCT through program POCI 2010 - 10-46 - IDMEC, subsidized by FEDER.

REFERENCES

[Jammes 2005] Jammes, F.; Harm, S., "Service-Oriented Paradigms in Industrial Automation, IEEE Transactions in Industrial Automation, Vol. 1, No.1, February 2005, pp. 62-70.

[Cardeira 2006] C. Cardeira, A. W. Colombo, R. Schoop, "Wireless solutions for automation requirements", in ATP International – Automation Technology in Practice, Vol 2, September 2006, pp 51-58.

[Bepperling 2006] Bepperling, A.; Mendes, J.; Colombo, A.W.; Schoop, R.; Aspragathos, A., "A Framework for Development and Implementation of Web service-Based Intelligent Autonomous Mechatronics Components," Industrial Informatics, 2006 IEEE International Conference on , vol., no., pp.341-347, 16-18 Aug. 2006

[Trebilcock 2007] Bob Trebilcock, "Automatic guided vehicle (AGV) basics: Guiding AGVs", *Modern Materials Handling*, January 2007.

[Durrant-Whyte 2006] Durrant-Whyte, H.; Bailey, T. "Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms". *Robotics and Automation Magazine* 13: 99–110.

[Davison 2002] A.J. Davison and D.W. Murray. Simultaneous Localization and Map-Building Using Active Vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 865–880, 2002.

[Liu and Thrun 2003] Y. Liu and S. Thrun. Results for outdoor-SLAM using sparse extended information filters. In *IEEE International Conference on Robotics and Automation*, volume 1, pages 1227–1233, 2003.

[Sunderhauf et al., 2007] N. Sunderhauf, S. Lange, and P. Protzel. Using the Unscented Kalman Filter in Mono-SLAM with Inverse Depth Parametrization for Autonomous Airship Control. In *IEEE International Workshop on Safety, Security and Rescue Robotics, 2007. SSRR 2007*, pages 1–6, 2007.

[Barfoot, 2005] T.D. Barfoot. Online visual motion estimation using Fast-SLAM with SIFT features. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'05)*, pages 3076–3082, 2005.

[Blanc et al. 2005] G. Blanc, T. Chateau, Y. Mezouar, O.A. Aider, P. Martinet, L. Eck, V. Moreau, and A. Nadim. Implementation of a vision-based navigation framework on a house mobile robot prototype. In *International Conference on Advances in Intelligent Systems-Theory and Applications, AISTA'04,* volume 8, pages 15–18.

[Gracias et al., 2003] Gracias, NR and Van Der Zwaan, S. and Bernardino, A. and Santos-Victor, J., Mosaic-based navigation for autonomous underwater vehicles, IEEE Journal of Oceanic Engineering, volume 28, number 4, pages 609-624, 2003

[Harris and Stephens, 1988] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50. Manchester, UK, 1988.

[Thrun et al., 1999], S. Thrun, M. Bennewitz, W. Burgard, AB Cremers, F. Dellaert, D. Fox, D. Hahnel, C. Rosenberg, N. Roy, J. Schulte, et al. MINERVA: A second-generation museum tour-guide robot. In *1999 IEEE International Conference on Robotics and Automation*, volume 3, 1999.

[Cardeira and Sa da Costa, 2005] C. Cardeira and J. Sa da Costa. A low cost mobile robot for engineering education. *IECON 2005. 32nd Annual Conference of IEEE Industrial Electronics Society*, pages 2162–2167, 2005.