The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Combined Constraint Matching Algorithm for Stereo Visual Odometry based on Local Interest Points

P. Núñez, R. Vázquez-Martín, A. Bandera, and F. Sandoval, *Member, IEEE*

***Abstract*— In this paper, we describe a new approach which uses scale-invariant image features to estimate the motion of a stereo head. These point features are matched between pairs of frames and linked into image trajectories at video rate, generating what it is called visual odometry, i.e. motion estimates from visual input alone. With respect to previously proposed approaches, the main novelty of our proposal is that the matching between sets of features associated to stereo pairs and between sets of image features associated to consecutive frames are conducted by means of a fast combined constraint matching algorithm. Besides, the efficiency of the approach is increased by using a closed-form solution to estimate the final robot displacement between consecutive acquired frames. We have tested the proposed approach for navigational purposes in a real environment. Experimental results demonstrate the performance of the proposal.**

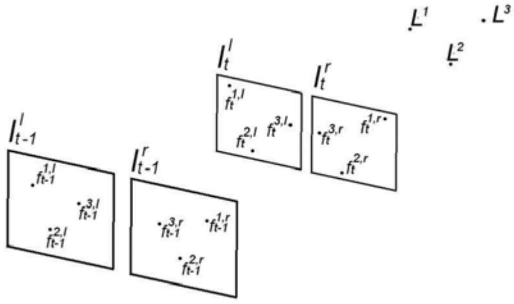

Fig. 1. Problem statement: given the pairs of stereo images taken at frames $t-1$ and t, the robot motion is estimated from the natural landmarks $\{L\}^i$.

I. INTRODUCTION

In order to accomplish higher-level tasks, autonomous mobile robots have to be able to determine their pose (position and orientation) while moving. In this sense, a precise and stable self-localization is one of the most important requirements to act purposefully in any environment. This task is typically performed using wheel odometry (from joint encoders) or inertial sensing (gyroscopes and accelerometers). However, wheel odometry techniques cannot be applied to robots with non-standard locomotion methods, such as legged robots. Besides, it suffers from precision problems, since wheels tend to slip and slide on the floor. On the other hand, inertial sensors are prone to drift. In robotics and computer vision, visual odometry defines the process of estimating the pose of a robot by analyzing the images provided by the camera(s) mounted on it [1]. Along the last decades, the visual odometry problem has widely been studied and there are successful techniques in the literature [2]–[5]. The most of these methods uses a stereo image pair matching, and estimate the ego-motion according to a set of corresponding points in two consecutive instant of time [2]–[4]. This matching process represents a crucial step for an accurate visual odometry method.

Specifically, the proposed visual odometry system consists of two consecutive matching stages (see Fig. 1). The first stage matches points of interest obtained from the left and right images, achieving stereo matching. These points are detected in both images by searching for maxima or minima of the Difference of Gaussians (DoG) function in the scale-space [6]. The scale-space representation is achieved by building an image pyramid with resampling in each level. For every point of interest, the Scale Invariant Feature Transform (SIFT) descriptor is calculated and stored with its scale and location values [6]. In spite of SIFT matching in stereo vision could be slower than using other type of features with identical results, it allows to obtain the set of SIFT for the next stage, which is interesting for its invariance to rotation and scale. Then, we perform stereo matching between points in the left and right images. This matching will be constrained by the stereo geometry -matched points must be in the same epipolar lines- and considering the scale and orientation of the descriptors. The aim is to provide a set of features which will be defined by their 3D world positions in the camera coordinate system. These features are considered as natural landmarks in the environment. Then, the second stage performs matching between sets of natural landmarks associated to consecutively acquired pairs of stereo images. This matching will be also constrained by the relative distance between features and the Euclidean distance of their SIFT descriptors. This stage allows to track the robot pose and its efficiency is achieved by a closed-form solution. Contrary to previous approaches which consider these matching stages as an outlier rejection process, the main novelty of this proposal is that these stages are carried out as an inlier detection process, which improves the accuracy of the matching process. Thus, both matching stages are stated as a maximum clique problem, i.e. to find the largest number of adjacent nodes for a given graph. In our case, nodes and arcs of this graph will be set according to the features to match.

This work has been partially granted by the Spanish Ciencia y Tecnologa (MCYT) and FEDER funds Project n. TIN2008-06196 and by the Junta de Andalucía Project n. P07-TIC-03106.

P. Núñez is member of the ISIS Group, and is with Dept. Tecnología de los Computadores y las Comunicaciones, Universidad de Extremadura, Spain (e-mail: pmnt@uma.es).

Rest of authors are with the Grupo de Ingeniería y Sistemas Integrados (ISIS), Dept. Tecnología Electrónica, Universidad de Málaga, Spain.

This paper is organized as follows: after discussing the main aspects of the maximum clique problem in Section II, Section III describes our method for visual odometry. Finally, experimental results and the main conclusions and future works are presented in Section IV and V, respectively.

II. Maximum Clique Problem

Let $G = (N, E)$ be an undirected graph with node set $N = \{n_1, ...n_n\}$. Two nodes n_i and n_j are said to be adjacent if they are connected by an arc $e_{ij} \in E$. A clique of a graph is a set of nodes where all of them are adjacent. Cliques with the following two properties have been studied over the last decades: *maximal cliques*, whose nodes are not a subset of the nodes of a larger clique, and *maximum cliques*, which are the largest among all cliques in a graph (maximum cliques are clearly maximal). In the maximum clique problem, one desires to find one maximum clique of an arbitrary undirected graph. This problem is computationally equivalent to some other important graph problems, for example, the maximum independent (or stable) set problem and the minimum node cover problem. Since these are NP-hard problems, no polynomial time algorithms are expected to be found.

In this work, we employ the branch-and-bound fast algorithm for the maximum clique problem proposed by [7]. Let $\{n\}_{i=1}^{n}$ be the set of nodes of the graph G and S_i be the subset $\{n_i, n_{i+1}, ...n_n\}$. The maximum clique algorithm firstly looks for cliques in S_n that contain n_n (the largest such clique is $\{n_n\}$), then cliques in S_{n-1} that contain n_{n-1}, and so on. The algorithm is presented in Table I. The set of nodes adjacent to a node n_i is denoted by $N(n_i)$ and the number of nodes in the graph is n. The global variable max gives the size of a maximum clique when the algorithm terminates.

The function $c(i)$ gives the largest clique in S_i. Obviously, for any $1 \leq i \leq n-1$, we have that $c(i) = c(i+1)$ or $c(i) = c(i+1)+1$. Moreover, we have $c(i) = c(i+1)+1$ iff there is a clique in S_i of size $c(i+1)+1$ that includes the node n_i. Therefore, starting from $c(n) = 1$, we search for such cliques. If a clique is found, $c(i) = c(i+1)+1$, otherwise $c(i) = c(i+1)$. The size of a maximum clique is given by $c(1)$. Old values of the function $c(i)$ enables the new pruning strategy (in line 14). That is, if we search for a clique of size greater than s, then we can prune the search if we consider n_i to become the $(j+1)$-th node and $j + c(i) \leq s$.

TABLE I

Fast Maximum clique algorithm [7]

```
function clique(U, size)
1: if |U| = 0 then
2:     if size > max then
3:         max := size
4:         New record; save it.
5:         found := true
6:     end if
7:     return
8: end if
9: while U ≠ ∅ do
10:    if size + |U| ≤ max then
11:        return
12:    end if
13:    i := min{j|n_j ∈ U}
14:    if size + c[i] ≤ max then
15:        return
16:    end if
17:    U := U\{n_i}
18:    clique(U ∪ N(n_i); size + 1)
19:    if found = true then
20:        return
21:    end if
22: end while
23: return
function new
24: max := 0
25: for i := n downto 1 do
26:    found := false
27:    clique(Si ∩ N(n_i), 1)
28:    c[i] := max
29: end for
30: return
```

III. Proposed Approach

A. Overview of the Proposal

The aim of the visual odometry algorithm is to calculate an estimate of the robot motion at each instant of time, Δp_t^r = $(\Delta x, \Delta y, \Delta\theta)^T$. In the proposed approach, a stereo camera mounted on the robot is used to provide a stereo view of the environment. Then, two consecutive image pairs acquired by the cameras are matched to estimate the displacement of the mobile platform. The quality of this matching process is crucial for accurate estimation. Indeed, a poor association between consecutive images leads to a robot pose error which is unrecoverable in most situations. Thus, a significant advance in visual odometry algorithms is the possibility of improving the matching process using consecutive stages [3]. The approach described in this paper follows this scheme, whose block diagram is illustrated in Fig. 2. As is shown in the figure, the proposed visual odometry algorithm consists of two matching processes which are achieved in five consecutive stages. First, each new image pair is acquired and the DoG detector is employed to obtain the two sets of points of interest [6]. These points are described using the SIFT algorithm [6]. Both sets of features are the input of the next stage, which computes the stereo matching. The 3D locations of these natural landmarks in the environment are calculated in the third stage using the output of this first stereo matching process. Next, the feature association stage performs matching between sets of features which belong to consecutively acquired stereo images. Finally, the output of this stage allows the system to estimate the robot displacement at current instant of time. Each one of these stages are explained in detail in the next subsections.

B. Scale-invariant Image Features

The Scale Invariant Feature Transform (SIFT) is a well-known method to provide a set of keypoints detected in the scale-space which are characterized by a descriptor invariant to scale and orientation. [6]. Let ${I^r}_t$ and ${I^l}_t$ be the right and left images captured using the stereo camera at time t. This first stage detects the set of SIFT features in two both

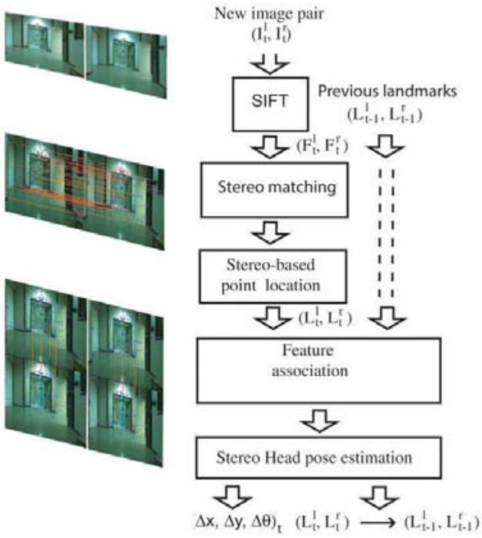

Fig. 2. Visual odometry algorithm proposed in this paper (see the text for a complete description of the modules).

images, ${F^l}_t$ and ${F^r}_t$, left and right, respectively. To aid the extraction of these features, the SIFT algorithm applies a four stage filtering approach to each image:

- *Scale-space extrema detection*. A scale spaced is constructed from the original image to determine those locations and scales that are identifiable from different views of the same object. To do this, the image is convolved with Gaussian filters at different scales, and then the difference of successive Gaussian-blurred images are calculated. Keypoints are then taken as maxima/minima of the Difference of Gaussians (DoG) in multiple scales.
- *Keypoint localization*. Previous stage produces too many keypoints candidates, some of which are unstable. Thus, this step treats to perform a detailed fit to the nearby data for accurate location, scale, and ratio of principal curvatures
- *Orientation assignment*. One or more consistent orientations are assigned to each keypoint based on local image gradients.
- *Keypoint descriptor.* Finally, a SIFT descriptor is generated for each keypoints from local image gradient information at the scale found in stage 2. This results in a feature vector containing 128 elements.

Figure 3 shows the SIFT features that were obtained from one pair of stereo images acquired with our vision system. A vector which represents the location, scale and orientation is associated with each descriptor. The scale and orientation of each feature is indicated by the size and orientation of this corresponding vector. The number of SIFT descriptors found in the stereo image shown in Fig. 3 was 94 for the left image, and 92 for the right image. Typically, the major problem of this SIFT feature description is the long time taken to extract the features from the images. Recent techniques reduce this computational load using the Iterative-SIFT algorithm [11],

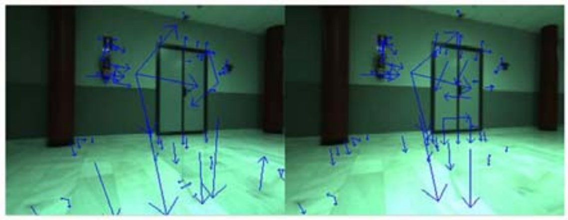

Fig. 3. SIFT features found for the left and right images from the stereo image (${F^l}_t$ and ${F^r}_t$). The scale and orientation are indicated by the size and orientation of the vectors.

or removing the rotational normalization and rotational part of the SIFT feature [12]. In our work we have limited the number of SIFT for each image in order to improve the processing time of the description stage.

C. Stereo Matching

In this section, we formulate the stereo matching problem as a graph-theoretic data association problem. The main advantage of our method respect to other stereo matching approaches is its robustness in the data association stage, improving the ego-estimation of the robot motion. The fundamental data structure of this stage is the correspondence graph [9], which represents valid associations between the two SIFT descriptor sets (see Fig. 4). Complete subgraphs or cliques within the graph indicate mutual associations compatibility and, by performing maximum clique search, the largest joint compatible association set may be found. Construction of the correspondence graph is performed through the application of relative and absolute constraints over the set of descriptors. Thus, the nodes of the graph indicate individual association compatibility and are determined by absolute constraint. On the other hand, the arcs of the correspondence graph indicate joint compatibility of the connected nodes and are determined by relative constraints. The method used to calculate the correspondence graph has three major steps:

1) *Nodes of the correspondence graph detection*. In the proposed method, graph nodes are associated to possible pairwise of SIFT features in ${F^l}_t$ and ${F^r}_t$ after applying an absolute constraint. Let $N_{{F^l}_t}$ and $N_{{F^r}_t}$ be the number of SIFT descriptors for left and right images, respectively. In a first place, the matrix T_t ($N_{{F^l}_t}$ x $N_{{F^r}_t}$) is generated for all these pairwise combinations calculating the Euclidean distance between these two descriptors. Therefore, two similar SIFT features in the image pair have low values in the matrix T_t. On the other hand, high values correspond to dissimilar features. The matrix is after modified to satisfied the constraint described in [10] (epipolar, disparity, orientation, scale and unique match constraints). The set of pairwise matched features whose matrix value is lower that a fixed threshold U_T constitutes the set of susceptible matched to be a real match in the stereo image. Thus, graph nodes are generated as all the possible combinations of these pairwise descriptors

(e.g. node 1 in Fig. 4 is valid if descriptor $F^{1,l}{}_t$ is a possible correspondence of $F^{1,r}{}_t$).

2) *Arcs of the correspondence graph construction*. For all pairwise combinations of matches in T_t, a relative constraint matrix is calculated, R_t. To do that, a relative constraint on the image coordinates is used. Let ω be the vector defined by $\omega = (o, s)^T$, where o and s are the orientation and scale values associated to the descriptor. Thus, a pair of matched descriptors is consistent if the Euclidean distance between the ω vectors from two SIFT descriptors in the left image is similar to the Euclidean distance between the corresponding vector in right image. That is, a pair of matches $(f^{i,l}{}_t,\ f^{i,r}{}_t)$ and $(f^{j,l}{}_t,\ f^{j,r}{}_t)$ are consistent iff they satisfy the relative constraint:

$$\|\omega_t^l - \omega_t^r\| \leq U_R^t \tag{1}$$

being

$$\begin{aligned} \omega_t^l &= \sqrt{(o_t^{i,l} - o_t^{j,l})^2 + (s_t^{i,l} - s_t^{j,l})^2} \\ \omega_t^r &= \sqrt{(o_t^{i,r} - o_t^{j,r})^2 + (s_t^{i,r} - s_t^{j,r})^2} \end{aligned} \tag{2}$$

where $(o,\ s)_i$ and $(o,\ s)_j$ denote the orientation and scale values of a SIFT descriptor and U_R^t is a threshold defined by the user. Thus, the corresponding entry in the relative constraint matrix R_t contains a 1 value if the constraint is satisfied (arc in the graph), and 0 otherwise (e.g. if the relative constraint between $(f^{1,l}{}_t,\ f^{2,l}{}_t)$ and $(f^{1,r}{}_t,\ f^{2,r}{}_t)$ matches, the node 1 is connected to the node 5 in Fig. 4).

3) *Maximum clique detection.* Next, the largest set of mutually consistent matches is calculated. This is equivalent of finding the maximum clique on a graph with adjacency matrix R_t. This problem was explained with detail in Section II. After applying the maximum clique algorithm, this stage provides a set of mutually compatible associations, that is, a set of matched SIFT features. Thus, the algorithm avoid bad associations, which results in erroneous displacement estimate. Figure 5 shows the pairwise SIFT descriptors after using the proposed stereo matching algorithm. As is illustrated in the figure, the quality of the matching process is guaranteed even though the number of SIFT descriptors is high. In this example, the number of matched features was 21.

D. Stereo-based Point Location

Each detected feature is readily characterized by the Cartesian localization of the point of interest provided by the stereoscopic vision system. Given the two matched features, the stereo vision system allows to compute the corresponding 3D point coordinates according to

$$Z = \frac{b \cdot f_c}{d} \qquad X = f_c\frac{(u - C_x)}{d} \qquad Y = \frac{f_c(v - C_y)}{d} \tag{3}$$

(C_x, C_y), b and f_c being the image center, stereo camera baseline and camera focal length, respectively. All of them

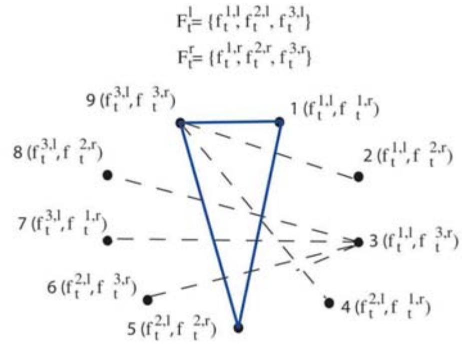

Fig. 4. Nodes represent possible matches when considered individually. Arcs indicate consistent associations, and a clique is a set of mutually consistent associations (e.g., the clique 1, 5, 9 implies associations $f^{1,l}{}_t \rightarrow f^{1,r}{}_t$, $f^{2,l}{}_t \rightarrow f^{2,r}{}_t$, $f^{3,l}{}_t \rightarrow f^{3,r}{}_t$ may coexist).

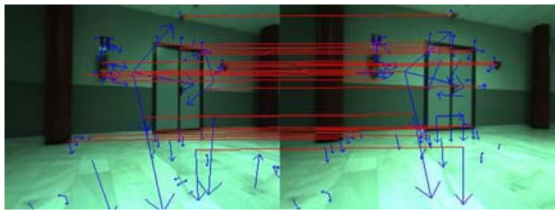

Fig. 5. Matched SIFT features between left and right images from the stereo pair shown in Fig. 3. Red line represents matched points.

are calibration parameters. Finally, d and (u, v) are the disparity value and the image coordinates location associated to the image feature assuming rectified images.

E. Feature Association

Let $I_{t-1}^{l,r}$ and $I_t^{l,r}$ represent the pairs of stereo images taken with the robot camera at two consecutive intervals of time. For each pair of images we detect points of interest, compute SIFT descriptors for them and perform stereo matching, resulting in two sets of natural landmarks L_{t-1} and L_t.

Next, we obtain the feature matching using a graph-theoretic data association problem. Thus, the correspondence problem is achieved between the set of landmarks associated to consecutive frames applying absolute and relative constraint (see Sec. III-C). The Euclidean distance between consecutive SIFT descriptors associated to this set of landmarks is used to obtain the matrix T_F. Thus, entries in T_F whose value are lower than a fixed threshold U_T constitute the set of possible matched landmarks. Next, the relative constraint is used to generate the adjacency matrix R_f from the set of possible pairwise landmarks. The relative constraint is changed for using the location of each pair of landmarks, $(L_{t-1}^i,\ L_{t-1}^j)$ and $(L_t^i,\ L_t^j)$. Thus, (1) is changed to

$$\|L_{t-1}^i - L_{t-1}^j\| - \|L_t^i - L_t^j\| \leq U_R^f \tag{4}$$

being $\|L_t^i - L_t^j\|$ the Euclidean distance between pair of landmarks using their 3D locations, and U_R^f an user-defined threshold. Finally, the maximum clique algorithm is applied to the adjacency matrix R_F and the set of mutually consistent

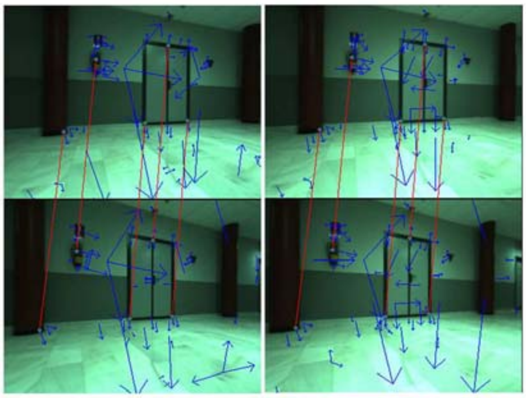

Fig. 6. Feature association results. After applying the maximum clique algorithm the number of pairwise matched features is 5 in both images.

matches is computed. Fig. 6 illustrates the features association results between two consecutive frames $t-1$ and t. The output of this stage provides a set of accurate pairwise matched features, which are used to obtain the displacement estimate. Let M denote the set of N_M landmarks matches, $\{(m_{t-1}^i,\, m_t^i)\}_{i=1:N_M}$, calculated by the proposed algorithm.

F. Stereo Head Pose Estimation

The purpose of the two-step matching process described in previous stage is to provide a set of good quality matches between consecutive frames which allows to calculate the estimated displacement of the robot. Typical solutions use an iterative method to estimate the robot displacement between two consecutive frames [3], [10]. In this paper, a closed solution is used to obtain an accurate and fast solution to the estimation problem. This solution is an adapted version for visual odometry application from novel scan matching approaches [8]. In spite this technique is able to be adapted for estimating 3D motions, in order to improve the computational load of the algorithm, the proposed method uses the projections to the ZX plane of the set of pairwise corresponded points. In other words, the coordinate Y of the interest point locations is omitted (we assume planar displacement). Thus, the algorithm minimizes the displacement error without iterations, thereby lessening the computational cost. The solution of this problem consists of minimizing the error function

$$E(R_{\Delta\theta}, \Delta T) = \sum_{i=1}^{N_M}\sum_{j=1}^{N_M} \eta_{ij} \left\| m_{t-1}^i - (R_{\Delta\theta} m_t^j + \Delta T) \right\|^2 \tag{5}$$

where m_{t-1}^i and m_t^j are matched interest points belonging to M, η_{ij} is a binary value defined as 1 if m_{t-1}^i and m_t^j have been matched or 0 otherwise, and $R_{\Delta\theta}$ and ΔT are the rotation and translation matrices whose values are sought. Thus, $\Delta p_t^r = (\Delta z,\ \Delta x,\ \Delta\theta)^T$ is the robot displacement estimated, that is, the output of the proposed odometry visual algorithm. An extended development of the optimization problem described in (5) is detailed in [8]. In that work, authors obtain a closed-form expression to estimate the robot motion.

IV. Experimental Results

To test the validity of the visual odometry algorithm, we use an ActiveMedia Pioneer 2AT robot mounted with an stereoscopic camera (see Fig.7a) and a 1.66GHz Pentium PC. The robot was driven through an indoor environment while capturing real-life stereo images. This real test is located at the research laboratories of the ISIS group in Málaga. The robot is teleoperated, and it moves across a corridor, closes a loop and returns to the starting location. Due to our space limitation for getting a longer experiment, this real test has been repeated several times with similar results. The stereo head is the STH-MDCS from Videre Design, a compact, low-power colour digital stereo head with an IEEE 1394 digital interface. The camera was mounted at the front and top of the vehicle at a constant orientation, looking forward. Images obtained were restricted to 320x240 pixels. Fig. 7b-g illustrates six different stereo captures from this real environment (each image in the figure represents the stereo pair at two consecutive frames, top and bottom of the image. The stereo matching and the feature matching is shown with red and yellow line, respectively). Besides, the robot pose at the capture time has been drawn over the trajectories (Fig. 8a).

The experimental results are focused on the accuracy and processing time of the proposed algorithm. The robot motion starts in the pose $p_{t=0}^r = (0, 0, 0°)^T$ and it was teleoperated across the environment, closing a loop and returning to the initial pose. The complete motion is constituted of 650 frames. This final location was measured with an uncertainty less than 2 mm in displacement and $0.2°$ in orientation. The wheel odometry is also saved and compared to the visual odometry. Fig. 8a shows the trajectories estimated by the proposed algorithm (red line) for this environment. Besides, the wheel odometry is included in the figure (green line). A zoom of the final pose is illustrated in the Fig. 8b. As is drawn in the figure, the visual odometry obtains an reliable estimate of the robot displacement improving the internal odometry at the end of the experiment. The robot pose estimate at the end of the experiment with the visual and wheel odometry are (800mm, 140mm, -2.15°)^T and (2392mm, 1105mm, 66.15°)^T, respectively. Finally, the average processing time of the algorithm is 420 ms (SIFT descriptors definition of the image pair: about 300ms, stereo matching: about 100 ms and feature matching: about 20ms). As is was commented in Section III-A, the most of the computational load is due to the description of the images using SIFT descriptors, which is able to be improve using a faster implementation of the algorithm (see Wu's implementation [13]). However, the experimental results demonstrate a satisfactory performance of the proposed approach.

V. Conclusions and Future Work

This paper has presented a combined constraint matching algorithm for a stereo visual odometry problem. In the

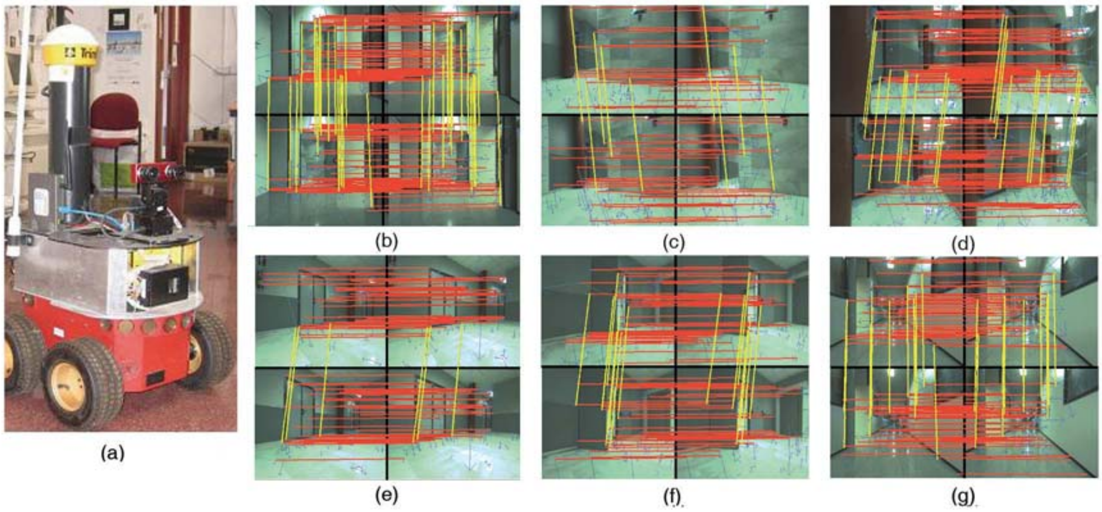

Fig. 7. a) Activmedia P2AT robot used in the experiments; b-g) six different stereo image pair captured across the robot motion. Stereo and feature matching are shown in the figure (red and yellow line, respectively).

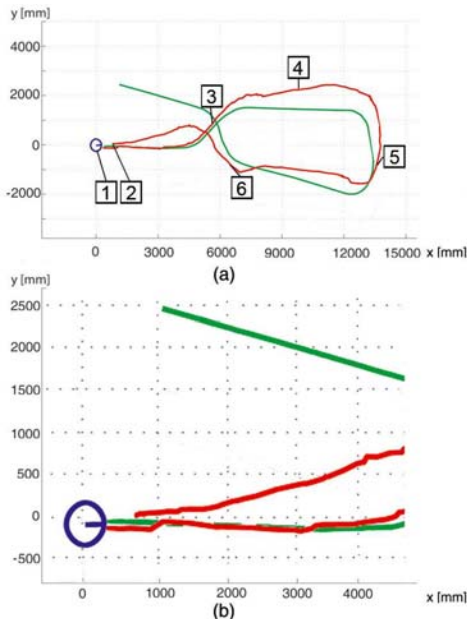

Fig. 8. a) Trajectories estimated by visual and wheel odometry (red and green line, respectively). Robot pose at the captured time shown in Fig. 7b-g are labeled; b) zoom of the robot pose at the end of the experiment.

matching process absolute and relative constraints have been integrated to achieved a multiple data tracking. The basis of the proposal is the use of local interest points detected from the image pairs. Our method provide a set of pairwise matched features in a two-step matching process, which are used to estimate the robot motion in a fast way. Thus, stereo visual odometry system proposed in this paper achieves an accurate estimate of the robot movement, improving the wheel odometry techniques. Our algorithm has been tested in a real scenario, demonstrating a high degree of reliability and accuracy.

Future works are focused on the implementation of a vision-based simultaneous localization and map building (vision-SLAM) algorithm which includes the visual odometry in the prediction stage. In order to reduce the computational load, the SIFT descriptors will be extracted using a implementation over the graphic hardware [13].

REFERENCES

[1] D. Nister, O. Naroditsky, and J. Bergen, "Visual odometry", in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (CVPR 2004), pp. 652-659, 2004.

[2] A. Mallet, S. Lacroix and L. Gallo, "Position Estimation in Outdoor Environments using Pixel Tracking and Stereovision" in Proc. IEEE Int. Conf. on Robotics and Automation, pp. 3519–3524, 2000.

[3] A. Howard, "Real-time stereo visual odometry for autonomous ground vehicles", in *Proc. IEEE Int. Conf. on Intelligent Robots and Systems*, pp. 3946–3952, 2008.

[4] J.P. Tardif, Y. Pavlidis and K. Daniilidis, "Monocular Visual Odometry in Urban Environments Using an Omnidirectional Camera", in *Proc. IEEE Int. Conf. on Intelligent Robots and Systems*, pp. 2531-2538, 2008.

[5] H. Hirschmuller, P. Innocent and J. Garibaldi, "Fast, Unconstrained Camera Motion Estimation from Stereo without Tracking and Robust Statistics", in , pp. 1099-1104, 2002.

[6] D.G. Lowe, "Object recognition from local scale-invariant features", in *Int. Conf. on Computer Vision*, Corfu–Greece, pp. 1150-1157, 1999.

[7] P. Östergard, "A fast algorithm for the maximum clique problem", in *Discrete Applied Mathematics*, 120, pp. 197-207, 2002.

[8] P. Núñez, R. Vázquez-Martín, A. Bandera, y F. Sandoval, "Fast laser scan matching approach based on adaptive curvature estimation for mobile robots". Accepted in *Robotica*, June 2008.

[9] H. G. Barrow and R. M. Burstall, "Subgraphs isomorphism, matching relational structures and maximal cliques", *Information Processing Letters*, 4, pp. 83–84, 1976.

[10] S. Se, D. Lowe and J. Little, "Mobile robot localization and mapping with uncertainty using scale invariant visual landmark", *Int. J. of Robotics Research*, 21, 8, pp. 83–84, 735-758.

[11] H.Tamimi, H. Andreasson, A. Treptow, and A. Zella, "Localization of mobile robots with omnidirectional vision using Particle Filter and iterative SIFT", *Robotics and Autonomous Systems*, pp. 758-765, 2006.

[12] L. Ledwich, S. Williams, "Reduced SIFT features for image retrieval and indoor localisation", in *Australasian Conf. on Robotics and Automation*, Canberra, 2004.

[13] C. Wu, "SiftGPU: A GPU Implementation of Scale Invariant Feature Transform (SIFT)", in http://cs.unc.edu/ ccwu/siftgpu, 2007.