Large Scale 6DOF SLAM with Stereo-in-Hand

***Abstract*—In this paper we describe a system that can carry out SLAM in large indoor and outdoor environments using a stereo pair moving with 6DOF as the only sensor. Unlike current visual SLAM systems that use either bearing-only monocular information or 3D stereo information, our system accommodates both monocular and stereo. Textured point features are extracted from the images and stored as 3D points if seen in both images with sufficient disparity, or stored as inverse depth points otherwise. This allows to map both near and far features: the first provide distance and orientation, and the second orientation information. Unlike other vision only SLAM systems, stereo does not suffer from 'scale drift' because of unobservability problems, and thus no other information such as gyroscopes or accelerometers is required in our system. Our SLAM algorithm generates sequences of conditionally independent local maps that can share information related to the camera motion and common features being tracked. The system computes the full map using the novel Conditionally Independent Divide and Conquer algorithm, which allows constant time operation most of the time, with linear time updates to compute the full map. To demonstrate the robustness and scalability of our system, we show experimental results in indoor and outdoor urban environments of $210m$ and $140m$ loop trajectories, with the stereo camera being carried in hand by a person walking at normal walking speeds of 4-5km/hour.**

***Index Terms*—Visual SLAM, Stereo, Scalability.**

I. INTRODUCTION: STATE OF THE ART IN VISUAL SLAM

THE interest in using cameras in SLAM has grown tremendously in recent times. Cameras have become much more inexpensive than lasers, and also provide texture rich information about scene elements at practically any distance from the camera. 6DOF SLAM systems based on 3D laser scanners plus odometry have been demonstrated feasible both indoors and outdoors [1], [2], as well as vision aided by laser without odometry [3] and vision aided by an inertial navigation system [4], [5]. But in applications where it is not practical to carry heavy and bulky sensors, such as egomotion for people tracking and environment modeling in rescue operations, cameras seem the only light weight sensors that can be easily adapted to helmets used by rescuers, or simply worn.

Current Visual SLAM research has been focused on the use of either monocular or stereo vision to obtain 3D information from the environment. Quite a few monocular visual SLAM systems have been demonstrated to be viable for small environments [6], [7], [8], [9], [10], [11], [12], [13], [14], [15]. Most are essentially standard EKF SLAM systems, and vary in the technique used to initialize a feature, given the partiality of the bearing only information provided by one camera, or in the type of interest points extracted from the images (be it Harris corners, Shi-Tomasi corners, SIFT features, or some combination). Some works have also considered segment features [16], [17]. Larger environments have been tackled in Hierarchical Visual SLAM [18].

A single camera is used in all of these systems, and although very distant features are potentially detectable, scale unobservability is a fundamental limitation. Either the scale is fixed in some way (for example by observing a known object [15]), or drift in scale can occur as is reported in the Hierarchical Visual SLAM system [18]. Panoramic cameras are also being used in visual SLAM [19], [20]. Here the limitation of scale unobservability is overcome using an additional stereo vision bench for motion estimation between consecutive frames. In the work of Royer et. at. [21] only monocular images are used. Mapping is achieved using a batch hierarchical bundle adjustment algorithm to compute all camera as well as interest points locations. The scale is introduced in the system by manually entering the length of the path.

Stereo visual systems provide scale through the baseline between the cameras, known from calibration. Davison and Murray demonstrated the first active stereo visual SLAM system [22], [23], [24]. It is based on standard EKF and thus also has low scalability. Under restrictive planar environment assumptions, Iocchi et. al. built an environment map using stereo [25]. Se et. al. demonstrated a visual stereo SLAM system using SIFT features in a small laboratory environment [26]. This system is also unlikely to scale adequately to large environments or work in more challenging outdoor scenarios as cross-correlations were neglected for computational reasons. In [27], [28] the authors demonstrate an autonomous blimp system for terrain mapping using stereo as the only sensor, also using a standard EKF SLAM algorithm. Saez et. al. [29] presented a 6DOF stereo visual SLAM system where egomotion estimation is done by a 3D point matching algorithm, and mapping through a global entropy minimization algorithm in

indoor orthogonal scenarios, with difficult extension to more complex non-orthogonal environments.

In [30], [31] Sim et. al. describe a dense visual SLAM system using Rao-Blackwellized Particle Filters and SIFT features (a similar effort in using Rao-Blackwellized Particle Filters and SIFT features for visual SLAM was reported in [14]). Visual odometry (SFM) is used to generate proposals for the sensor motion and global pose estimation algorithms for loop closing. This system works in either monocular or stereo mode, with cameras mounted on a robot moving in 2D; sensor trajectories with 6DOF will require large amounts of particles for their representation. In [32] the authors also compare the advantages of separate monocular and stereo approaches in traditional SLAM frameworks.

In this paper we show the advantages of being able to accommodate both monocular and stereo information in carrying out 6DOF SLAM with a hand-held camera. In the works of Sola et. al. [33] and Lemaire et. al. [19] it is also pointed out that combining visual information at close range as well as at infinity should improve the performance of visual SLAM.

Since the initial results of [34] great progress has been made in the related problem of visual odometry [35], [36], [37], [38]. Visual odometry systems have the important advantage of constant time execution. Furthermore, during exploratory trajectories, in which an environment feature is seen for a certain window of time and never more, visual odometry can obtain the same precision in the estimation of the sensor location as a SLAM system, with a great reduction in cost. Unfortunately, visual odometry does not cope with loop closings, and thus eventual drift in these cases is inevitable. Stereo visual odometry combined with GPS can result in a mapping system that avoids long term drift [39], [40], but unfortunately GPS is not always available. Improving the precision in sensor location through loop closing is one of the main advantages of SLAM.

An important limitation of current SLAM systems that use the standard EKF algorithm is that when mapping large environments, very soon they face computational as well as consistency problems [41], [42]. Many efforts have been invested in reducing the $O(n^2)$ cost of the EKF updates. In [43] an Information filter, the dual of the Kalman Filter, was used, allowing constant time updates irrespective of the size of the map. An approximation is carried out to sparsify the Information Matrix, which may lead to map divergency [44]. The Tree Map algorithm [45] performs updates in $O(\log n)$ also by forcing information matrix sparseness by weak link breakage. In more complicated trajectories, such as lawnmowing, the cost can be more than log linear [46]. In the Smoothing and Mapping method [47], the authors observed that the information matrix is exactly sparse when all vehicle locations are considered in the stochastic map, and thus very efficient techniques can be used to compute the batch solution (a recent incremental version is described in [48]).

All of these algorithms use the information form and thus the state and covariance are not readily available. There are alternatives that work on the covariance form, such as the Map Joining Algorithm [49]. It works on a sequence of local maps of limited size and thus it can cut down the cost of EKF SLAM considerably, although remaining $O(n^2)$. It has the additional advantage of improving the consistency of the resulting estimation [42]. The Divide and Conquer algorithm [50], [51] is able to compute the covariance form of the stochastic map in amortized time linear with the size of the map, improving further the consistency of the solution. However, in these systems, local maps are required to be statistically independent. This requires creating a new local map from scratch every time the current local map size limit has been reached. Consequently, no sharing is possible of valuable information in 6DOF visual SLAM, such as the camera velocity, or information about features currently being tracked.

In this paper we describe a robust and scalable 6DOF visual SLAM system that can be carried in hand at normal walking speeds of 4-5km/hour, and used to map large indoor and outdoor environments. In section II we summarize the main charateristics of our system. In section III we describe the details of the visual SLAM system that provides the sequence of conditionally independent local maps; the basic building blocks of our mapping algorithm. This algorithm, Conditionally Independent Divide and Conquer SLAM, is explained in section IV. In section V we describe the two experiments carried out to test the system, an indoor 200m loop and an outdoor 140m loop. In section VI we discuss the results obtained, and finally in section VII we draw the main conclusions of our work.

II. Our proposal

The fundamental characteristics of the system that we describe in this paper are:

1) Unlike any other visual SLAM system, we consider information from features both close and far from the cameras. Stereo provides 3D information from nearby scene points, and each camera can also provide bearing only information from distant scene points. Both types of information are incorporated into the map and used to improve the

Fig. 1. Stereo vision system used to acquire the image sequences. Picture on the left shows the experimental setup during the data acquisition for the indoor experiment.

estimation of both the camera pose and velocity, as well as the map.

2) Nearby scene points provide scale information through the stereo baseline, eliminating the intrinsic scale unobservability problem of monocular systems.
3) We use Conditionally Independent Divide and Conquer SLAM, a novel SLAM algorithm that allows to maintain both camera velocity information and current feature information during local map initialization. This adds robustness to the system without sacrificing precision or consistency in any way. Being a Divide and Conquer algorithm, it also allows linear time execution, enabling the system to be used for large scale indoor/outdoor SLAM.

Our 6DOF hardware system consists of a stereo camera carried in hand and a laptop to record and process a sequence of images (fig. 1). Since the camera moves in 6DOF, we define the camera state using 12 variables: camera position in 3D cartesian coordinates, camera orientation in Euler angles, and linear and angular velocities. It is known that a stereo camera can provide depth estimation of points up to a certain distance determined by the baseline between left and right cameras. Therefore, two regions can be differentiated: a region close to the cameras and visible by both, in which stereo behaves as a range and bearing sensor. The second is the region of features far from the cameras or seen by only one, in which the stereo becomes a monocular camera, only providing bearing measurements of such points. To take advantage of both types of information, we combine 3D points and inverse depth points (introduced in [52]) in the state vector in order to build a map and estimate the camera trajectory. The system produces sequences of local maps of limited size containing both types of features using an EKF SLAM algorithm. As we detail in section IV, these local maps are joined into a full map using the Conditionally Independent Divide and Conquer SLAM algorithm, obtaining as final result a full stochastic map containing all tracked features and the final and intermediate camera states from each local map. This system is highly scalable: local maps are built in constant time, regardless of the size of the environment, and the Conditionally Independent Divide and Conquer algorithm requires amortized linear time.

During the feature tracking process, the right image is chosen as reference to initialize new features. Interest points are detected and classified according to their disparity with the left image. Those points whose disparity reveals a close distance are initialized as 3D features, otherwise they are modeled as inverse depth points and initialized using the bearing information obtained from the right image. When the camera moves, these features are tracked in order to update the filter and produce the corresponding corrections. To track a feature, its position is predicted in both images inside a bounded region given by the uncertainty in the camera motion and the corresponding uncertainty of the feature.

The process to select, initialize, and manage these features is detailed in the next section.

III. The Visual SLAM System

A. State Representation

The state vector that represents a local submap $\mathbf{x}_B$ contains the final camera location $\mathbf{x}_c$ and the location of all features $\mathbf{x}_{f_{1:n}}$ with respect to the map base reference B, the initial camera location. Some features are codified using the *Inverse Depth (ID) parametrization* that model points that are at the infinity in $\mathbf{x}_{ID}$. Additionally, cartesian $3D$ *parametrization* is used to represent depth points in $\mathbf{x}_{3D}$:

$$\mathbf{x}_B = \begin{bmatrix} \mathbf{x}_c \\ \mathbf{x}_{f_{1:n}} \end{bmatrix} = \begin{bmatrix} \mathbf{x}_c \\ \mathbf{x}_{ID} \\ \mathbf{x}_{3D} \end{bmatrix} \tag{1}$$

The camera is described by the position of its optical center in cartesian coordinates $\mathbf{r}$, its orientation in Euler angles Ψ, its linear velocity $\mathbf{v}$ and its angular velocity $\mathbf{w}$. In order to carry out the prediction process, the camera motion follows a constant velocity model with zero mean Gaussian noise in the linear and angular accelerations:

$$\mathbf{x}_c = \begin{bmatrix} \mathbf{r} \\ \Psi \\ \mathbf{v} \\ \mathbf{w} \end{bmatrix} \tag{2}$$

Image corners classified as depth points are transformed to 3D points, given the disparity information provided by the stereo pair. Subsection III-D describes the criterion adopted to select points as depth points. Since

the stereo camera provides rectified images, the back-projection equations to obtain a 3D point are based on a pinhole camera model which relates image points and 3D points using the following transformation function:

$$\begin{aligned}\mathbf{x}_{3D} &= f(u_r, v_r, u_l, v_l) \\ &= [x, y, z]^T \\ &= \left[\frac{b(u_r-u_0)}{d}, \frac{b(v_r-v_0)}{d}, \frac{fb}{d}\right]^T \end{aligned} \quad (3)$$

where (u_r, v_r) and (u_l, v_l) are the pixels on the right and left images, and $d = (u_l - u_r)$ is the horizontal disparity. The remainder terms in the equations are the calibrated parameters of the camera, i.e., the central pixel of the image (u_0, v_0), the baseline b and the focal length f.

Given the camera location $\mathbf{x}_{c_i}$, an inverse depth point is defined as in [52]:

$$\mathbf{x}_{ID} = \begin{bmatrix} \mathbf{r}_i \\ \theta_i \\ \phi_i \\ \rho_i \end{bmatrix} \quad (4)$$

This vector depends on the optical center $\mathbf{r}_i$ of the camera from which the feature was first observed, the direction of the ray passing through the image point (i.e. azimuth θ_i, elevation ϕ_i), and the inverse of its depth, $\rho_i = 1/d_i$.

B. Selection and Management of Trackable points

To ensure tracking stability of map features, distinctive points have to be selected. Following a similar idea as the one presented in [53], we use the Shi-Tomasi variation of the Harris corner detector to select good trackable image points and their corresponding 11×11 surrounding patch.

From the first step, the right image is split using a regular grid; the point with the best detector response per grid cell is selected, see fig. 2. At each step, we use only those features that fall in the FOV of the camera when they are projected along with their uncertainties on right and left images. Using the patch associated with each feature, a matching search based on normalized cross-correlation is performed inside the projected uncertainty region, as introduced in [23]. During the following steps those cells that become and remain empty for a given time are monitorized to initialize a new feature when a good point is detected. In this way features can be uniformly distributed in the image, improving the amount of information gathered from the scene and therefore the map estimate. The approach is accompanied by a feature management strategy so that non-persistent features are deleted from the state vector to avoid an unnecessary growth in population.

C. Measurement Equation

At each step, features that fall in the field of view of each camera in the stereo are projected with their uncertainty to the corresponding camera image. Using the patch associated with the feature, a match inside the projected uncertainty region is searched using normalized cross-correlation. When a match is found, a new observation $\mathbf{z}$, given by the matched pixel is used to update the state of the camera and the map.

In the right camera, the equation that defines the relation between the ith inverse depth feature $\mathbf{x}^i_{ID}$ and its observation $\mathbf{z}^{r_i}_{ID}$ is given by the measurement equation:

$$\begin{aligned}\mathbf{z}^{r_i}_{ID} &= h^r_{ID}(\mathbf{x}_c, \mathbf{x}^i_{ID}) + \upsilon \\ &= projection(\ominus\mathbf{x}_c \oplus \mathbf{x}^i_{ID}) + \upsilon \end{aligned} \quad (5)$$

where h^r_{ID} is the function that projects the inverse depth feature to the right camera and υ is a zero mean gaussian noise with σ_p standard deviation that represents the projection error in pixels. Alternatively, we can define the measurement equation that relates the inverse point observation on the left image by:

$$\begin{aligned}\mathbf{z}^{l_i}_{ID} &= h^l_{ID}(\mathbf{x}_c, \mathbf{x}^i_{ID}) + \upsilon \\ &= projection(\ominus\mathbf{x}_c \oplus \mathbf{x}_{c_r c_l} \oplus \mathbf{x}^i_{ID}) + \upsilon \end{aligned} \quad (6)$$

where the displacement of the left camera optical center with respect to the right camera is given by the rigid transformation $\mathbf{x}_{c_r c_l} = [0\ b\ 0]^T$.

In a similar way, we describe observations corresponding to 3D map features in the right and left cameras:

$$\begin{aligned}\mathbf{z}^{r_i}_{3D} &= h^r_{3D}(\mathbf{x}_c, \mathbf{x}^i_{3D}) + \upsilon \\ &= projection(\ominus\mathbf{x}_c \oplus \mathbf{x}^i_{3D}) + \upsilon \\ \mathbf{z}^{l_i}_{3D} &= h^l_{3D}(\mathbf{x}_c, \mathbf{x}^i_{3D}) \\ &= projection(\ominus\mathbf{x}_c \oplus \mathbf{x}_{c_r c_l} \oplus \mathbf{x}^i_{ID}) + \upsilon \end{aligned}$$

Note that we use $\oplus$ and $\ominus$ operators in order to denote the corresponding compositions and inversions of transformations. They represent different transformations depending on the kind of parametrization used to express a feature. In [49], the definitions for 2D transformations were introduced, dealing mainly with point features and line features. In [54], the operations have been extended for 3D inverse depth and depth points. Details of the calculation of the corresponding Jacobians to propagate the uncertainties correctly can also be found in [54].

Fig. 2 shows the prediction of those 3D and inverse depth features that fall inside the field of view of each of the cameras. A good advantage of using a stereo camera

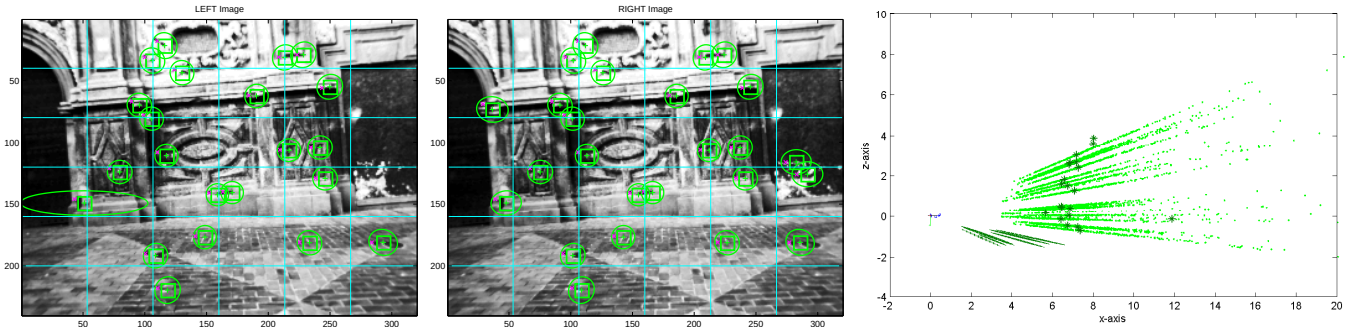

Fig. 2. Points detected using a stereo camera. Projection of map features on both left (left) and right (middle) images. We show feature uncertainties from a lateral perspective (right): 3D feature uncertainties are drawn using darker ellipses whereas we use samples to show the inverse depth feature uncertainties. The accompanying video `VSLAM_local_map.avi` illustrates the process of building a single local submap.

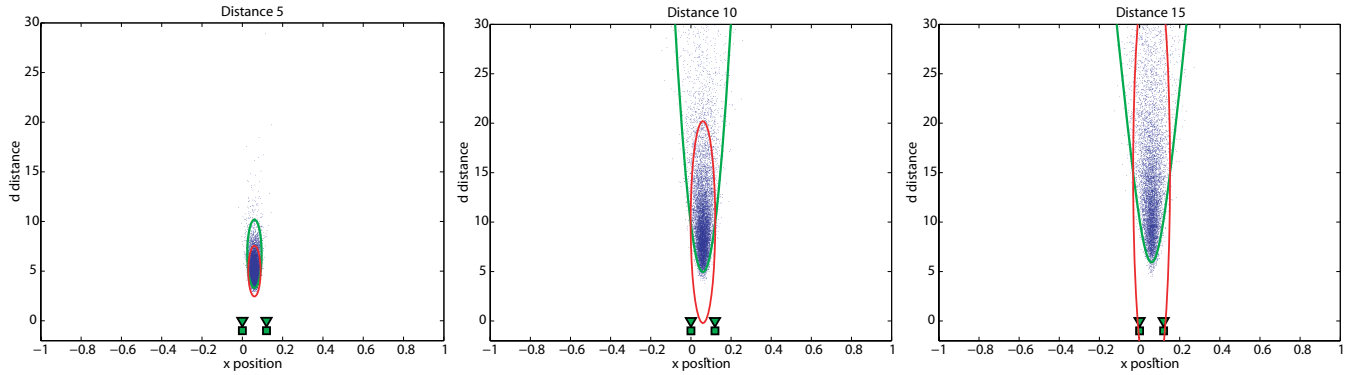

Fig. 3. Simulated experiment of a point reconstruction from a stereo pair observation, for a point at 5m distance (left), 10m (middle) and 15m (right). The point clouds are samples from the real distribution of the point location, given that the pixel noise in the images is Gaussian. Black ellipses represent the uncertainty region for the point location when the back projection equations of a depth point are linearized. Grey regions represent the uncertainty in the point using the inverse depth parametrization. The accompanying video `VSLAM_stereo_distribution.avi` shows the real and approximate uncertainties.

is that although a feature can disappear from the field of view of one camera, information to update the state is available if the feature can be still found in the other. As it will be shown in the experiments, this fact is of extreme importance when the camera rotates or turns around a corner, since features escape very fast from the FOV of a single camera making the estimation of the camera location in those moments very weak.

D. Depth points Vs. Inverse Depth points

Current research on Monocular SLAM has shown that the inverse-depth parametrization is suitable to represent the distribution of features at infinity as well as close points, allowing to perform an undelayed initialization of features. Despite its properties, each inverse depth point needs an over-parametrization of six values instead of a simpler three coordinates spatial representation [55]. This produces a computational overhead in the EKF. Working with a stereo camera, which can estimate the depth of points close to the camera, raises the subtle question of when a feature should be initialized using a $3D$ or an *ID* representation.

In order to clarify this issue we have designed a simulated experiment to study the effect of the linearization in both representations when a point is initialized using the stereo information. In this simulated experiments the variance of the pixel noise $(\sigma_p = 1pixel)$ and the actual intrinsic parameters of the stereo camera used, such as the baseline, are taken into account to implement the simulation. The experimental setup consists of a stereo pair where the left camera is located at the origin of the reference frame, with its principal axis pointing along Z and the X axis pointing to the right. The right camera is at $b = 12cm$ in X. We consider a point that is in the middle between both cameras at different distances in Z. Given a noisy pixel observation the uncertainty region of a reconstructed point is sampled and plotted in Fig.3 for three different point distances: 5,10 and 15 meters. The uncertainty region of the 3D representation which is calculated using a linearization of Eq. 3 and evaluated

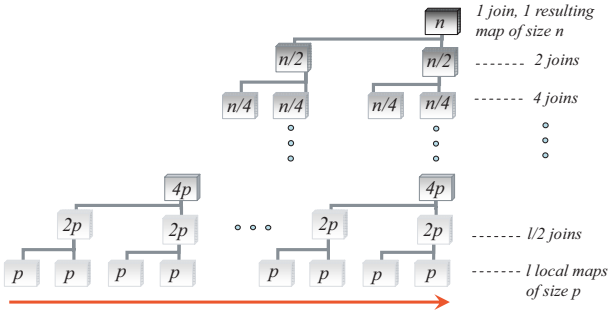

Fig. 4. Binary tree representing the hierarchy of maps that are created and joined in D&C SLAM. The red line shows the sequence in which maps are created and joined.

in the ground truth, is represented by the dark ellipse. The corresponding uncertainty region of the linearized inverse depth representation is bounded by the grey lines in the plot. Notice that the inverse depth parametrization models very accurately the real uncertainty for the studied distances. However, although the dark ellipse covers the real distribution at 5 meters quite accurately, for longer distances the ellipse overestimates the uncertainty in the region close to the cameras and is overconfident for far distances.

This empirical analysis suggest choosing a threshold of 5 meters. A point closer than $5m$ is initialized using a 3D representation, a more distant point is parameterized as an inverse depth point.

Inverse depth features can be transitioned to 3D points reducing significantly the number of DOF. Conversion requires an analysis of the linearity of the functions that models both depth point and inverse depth point distributions. In [55] this issue is considered by using a linearity index. Such analysis makes it possible to decide when an inverse point distribution is well approximated with the over parameterized coding. Switching from inverse depth to depth depends on a linearity threshold derived from the analysis.

IV. Conditionally Independent Divide and Conquer SLAM

Divide and Conquer SLAM (D&C) has proved to be a good algorithm in minimizing the computational complexity of EKF-based SLAM and improving consistency of the resulting estimate [50]. The algorithm allows to efficiently join several local maps into a single state vector using Map Joining in a Hierarchical tree structure (figure 4). Local maps can be obtained in constant time, regardless of the size of the environment, and the map joining operations can be performed in amortized linear time. The D&C SLAM algorithm was however conceived for statistically independent sequences of local maps. This requires creating a new local map from scratch every time the current local map size limit has been reached. Consequently, it is not possible to share valuable information in 6DOF visual SLAM, such as the camera velocity, or information about features currently being tracked.

In this section we describe the *Conditionally Independent* D&C SLAM algorithm, that is able to work with maps that are not statistically independent, but rather *conditionally independent*, and thus allow to share the valuable information with no increment in computational cost or loss of precision whatsoever.

A. Conditionally Independent Local Maps

In Visual SLAM it can be very useful to share some state vector components between consecutive submaps: some camera states, such as linear and angular velocities, as well as features that are in the transition region between adjacent submaps and are currently being tracked. This allows to improve the estimate of relative location between the submaps and continue tracking the observed features with no interruptions. Nevertheless, special care is needed to join the submaps in a single map since their estimates are not independent anymore.

The novel technique to achieve these requirements is based on the concept of Conditionally Independent Local Maps (CI) presented in [56]. For the reader convenience here we present a brief summary of the technique.

Suppose that a local map 1 has been built and we want to start a new submap 2 not from scratch, but sharing some elements in common with 1. Submap 1 is described by the following probability density function:

$$p(\mathbf{x}_A, \mathbf{x}_C|\mathbf{z}_a) = \mathcal{N}\left(\begin{bmatrix} \hat{\mathbf{x}}_{A_a} \\ \hat{\mathbf{x}}_{C_a} \end{bmatrix}, \begin{bmatrix} P_{A_a} & P_{AC_a} \\ P_{CA_a} & P_{C_a} \end{bmatrix}\right) \tag{7}$$

where $\mathbf{x}_A$ are the components of the current submap that only belong to map 1, $\mathbf{x}_C$ are the elements that will be shared with map 2, and $\mathbf{z}_a$ the observations gathered during the map construction. Notice that upper case subindexes are for state vector components whereas lower case subindexes describe which observations $\mathbf{z}$ have been used to obtain the estimate.

Submap 2 is then initialized with the result of marginalizing out the non common elements from submap 1:

$$p(\mathbf{x}_C|\mathbf{z}_a) = \int p(\mathbf{x}_A, \mathbf{x}_C|\mathbf{z}_a)\, d\mathbf{x}_A = \mathcal{N}(\hat{\mathbf{x}}_{C_a}, P_{C_a}) \tag{8}$$

During the trajectory along map 2, new observations $\mathbf{z}_b$ are gathered about the common components $\mathbf{x}_C$

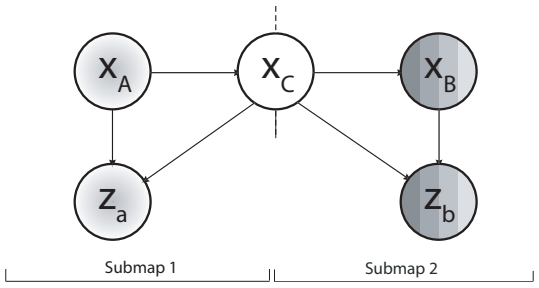

Fig. 5. Bayesian network that describes the relations between two consecutive submaps

as well as observations of new elements $\mathbf{x}_B$ that are incorporated to the map. When map 2 is finished, its estimate is finally described by:

$$p(\mathbf{x}_C, \mathbf{x}_B | \mathbf{z}_a, \mathbf{z}_b) = \mathcal{N}\left(\begin{bmatrix} \hat{\mathbf{x}}_{C_{ab}} \\ \hat{\mathbf{x}}_{B_{ab}} \end{bmatrix}, \begin{bmatrix} P_{C_{ab}} & P_{CB_{ab}} \\ P_{BC_{ab}} & P_{B_{ab}} \end{bmatrix} \right) \tag{9}$$

where the subindexes in the estimates $\hat{\mathbf{x}}_{C_{ab}}$ and $\hat{\mathbf{x}}_{B_{ab}}$ reveal that both sets of observations $\mathbf{z}_a$ and $\mathbf{z}_b$ have been used in the estimation process. This means that submap 2 is updated with all the information gathered by the sensor. But observe that map 1 in Eq. (7) has been updated with the observation $\mathbf{z}_a$ but not with the more recent observations $\mathbf{z}_b$.

Figure 5 shows a Bayesian network that describes the probabilistic dependencies between elements of submaps 1 and 2. As it can be seen, the only connection between the set of nodes ($\mathbf{x}_A$, $\mathbf{z}_a$) and ($\mathbf{x}_B$, $\mathbf{z}_b$) is through node $\mathbf{x}_C$, i.e. both subgraphs are *d-separated* given $\mathbf{x}_C$ [57]. This implies that nodes $\mathbf{x}_A$ and $\mathbf{z}_a$ are *conditionally independent* of nodes $\mathbf{x}_B$ and $\mathbf{z}_b$ given node $\mathbf{x}_C$. Intuitively this means that if $\mathbf{x}_C$ is known, submaps 1 and 2 do not carry any additional information about each other.

B. Conditionally Independent Map Joining

Consider two consecutive CI local maps. The operations to join the two maps into a single stochastic map that contains all the information provided by each map are as follows:

$$\begin{aligned} p(\mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C | \mathbf{z}_a, \mathbf{z}_b) = & \\ = \mathcal{N}\left(\begin{bmatrix} \hat{\mathbf{x}}_{A_{ab}} \\ \hat{\mathbf{x}}_{C_{ab}} \\ \hat{\mathbf{x}}_{B_{ab}} \end{bmatrix}, \begin{bmatrix} P_{A_{ab}} & P_{AC_{ab}} & P_{AB_{ab}} \\ P_{CA_{ab}} & P_{C_{ab}} & P_{CB_{ab}} \\ P_{BA_{ab}} & P_{BC_{ab}} & P_{B_{ab}} \end{bmatrix} \right) & \end{aligned} \tag{10}$$

Taking into account the submap conditional independence property, it can be demonstrated [56] that the optimal map result of the joining can be computed using:

$$\begin{aligned} K &= P_{AC_a} P_{C_a}^{-1} \\ &= P_{AC_{ab}} P_{C_{ab}}^{-1} && (11) \\ \hat{\mathbf{x}}_{A_{ab}} &= \hat{\mathbf{x}}_{A_a} + K(\hat{\mathbf{x}}_{C_{ab}} - \hat{\mathbf{x}}_{C_a}) && (12) \\ P_{A_{ab}} &= P_{A_a} + K(P_{CA_{ab}} - P_{CA_a}) && (13) \\ P_{AC_{ab}} &= K P_{C_{ab}} && (14) \\ P_{AB_{ab}} &= K P_{CB_{ab}} && (15) \end{aligned}$$

Using this technique, we can build local maps that have elements in common and then retrieve the global information in a consistent manner. After the joining, the elements belonging to the second map are transformed to the base reference of the first map.

C. Actual implementation for stereo

The D&C SLAM algorithm of [50] can be adapted to work with conditional independent local maps simply by using the CI Map Joining operation described above. As we mentioned before, since the camera moves in 6DOF, the camera state is composed of its position using 3D cartesian coordinates, the orientation in Euler angles and its linear and angular velocities. 3D points and inverse depth points are included as features in the state vector. When a local map $\mathbf{m}_i$ is finished, the final map estimate is given by:

$$\mathbf{m}_i.\hat{\mathbf{x}} = \begin{bmatrix} \hat{\mathbf{x}}_{R_i R_j} \\ \hat{\mathbf{v}}_{R_i R_j} \\ \hat{\mathbf{x}}_{R_i F_{1:m}} \\ \hat{\mathbf{x}}_{R_i F_{m+1:n}} \end{bmatrix} \tag{16}$$

where $\hat{\mathbf{x}}_{R_i R_j}$ is the final camera location R_j with respect to the initial one, R_i, $\hat{\mathbf{v}}_{R_i R_j}$ are the linear and angular velocities, $\hat{\mathbf{x}}_{R_i F_{1:m}}$ are 3D and inverse depth features that will only remain in the current map and $\hat{\mathbf{x}}_{R_i F_{m+1:n}}$ are 3D and inverse depth features that will be shared with the next submap $\mathbf{m}_j$.

Since the current camera velocity $\hat{\mathbf{v}}_{R_i R_j}$ and some features $\hat{\mathbf{x}}_{R_i F_{m+1:n}}$ are used to initialize the next local map, these elements have to be computed with respect to the base reference of the second map R_j:

$$\mathbf{m}_i.\hat{\mathbf{x}} = \begin{bmatrix} \hat{\mathbf{x}}_{R_i R_j} \\ \hat{\mathbf{v}}_{R_i R_j} \\ \hat{\mathbf{x}}_{R_i F_{1:m}} \\ \hat{\mathbf{x}}_{R_i F_{m+1:n}} \\ \cdots \\ \ominus \hat{\mathbf{x}}_{R_i R_j} \oplus \hat{\mathbf{v}}_{R_i R_j} \\ \ominus \hat{\mathbf{x}}_{R_i R_j} \oplus \hat{\mathbf{x}}_{R_i F_{m+1:n}} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{x}}_{A_a} \\ \cdots \\ \hat{\mathbf{x}}_{C_a} \end{bmatrix} \tag{17}$$

where the new elements define the common part $\hat{\mathbf{x}}_{C_a}$ and the original map defines $\hat{\mathbf{x}}_{A_a}$. Notice that the

appropriate composition operation have to be applied for each transformed component and that the corresponding covariance elements have to be added to the map.

In local mapping, a base reference has to be identified to start a new map. This common reference is represented by the final vehicle position, which is the case of R_j between $\mathbf{m}_i$ and $\mathbf{m}_j$.

The initial state vector of the next submap is then given by:

$$\mathbf{m}_j.\hat{\mathbf{x}} = \begin{bmatrix} \hat{\mathbf{x}}_{R_jR_j} \\ \ominus\hat{\mathbf{x}}_{R_iR_j} \oplus \hat{\mathbf{v}}_{R_iR_j} \\ \ominus\hat{\mathbf{x}}_{R_iR_j} \oplus \hat{\mathbf{v}}_{R_iR_j} \\ \ominus\hat{\mathbf{x}}_{R_iR_j} \oplus \hat{\mathbf{x}}_{R_iF_{m+1:n}} \end{bmatrix} \tag{18}$$

where $\hat{\mathbf{x}}_{R_jR_j}$ represents the location of the camera in the new reference frame with initial zero uncertainty and zero correlation with the rest of the elements of the initial map. Notice that the initial velocity brought from the previous map has been replicated twice. One of the copies will change as the camera moves through the new map carrying the current camera velocity. The other copy will remain fixed and, together with the transformed features, will be the common elements with the previous map. The same process is successively repeated with all local maps.

D. Continuous data association in each local map

Recent work on large environments [18] has shown that the Joint Compatibility Test [58] helps avoiding map corruption in visual SLAM by rejecting measurements that come from moving objects. This framework is suitable in environments with a limited number of observations. However, a Branch and Bound algorithm implementation of (**JCBB**) has limited use when the number of observations per step is large. In this paper we have obtained more efficient results using the *Randomized Joint Compatibility* version **RJC** proposed in [51], in which, in the spirit of RANSAC, a *Joint Compatibility* **JC** test is run with a fixed set of p randomly selected measurements. In this case, correlation between patches and individual χ^2 tests are used to obtain candidate matches. If all p measurements and their matches are jointly compatible, we apply the Nearest Neighbor rule to match the remaining measurements. Once a full hypothesis H is obtained, we check **JC** to avoid false positives. The process is repeated t times with adaptive RANSAC, limiting the probability of missing a correct association.

E. Map matching

The property of sharing common elements solves the data association problem between consecutive local maps [51]. This requires of us to solve data association only in loop closing situations. We use the Maximum Clique Algorithm of [18] in order to detect a previously visited area. The algorithm finds correspondences between features in different local maps, taking into account the texture and the relative geometry between the features. If sufficient corresponding features are found, an ideal measurement equation that imposes the loop closing constraint is applied in the final map.

V. Experiments in Urban Outdoor and Indoor Environments

In order to demonstrate de robustness and scalability of the visual SLAM system that we propose, we have gathered two 320x240 images sequences with a Point Grey Bumblebee stereo system (see fig. 1). The system provides a 65 x 50 degree field of view per camera, has a baseline of $12cm$, limiting the 3D point features initialization up to a distance close to $5m$.

An indoor loop (at 48 fps) and an urban outdoor (at 25 fps) loop sequences were captured carrying the camera in hand, at normal walking speeds of $4-5km/hour$. Both sequences were processed in MATLAB with the proposed algorithms on a desktop computer with an Intel 4 processor at 2,4GHz. The higher frame rate for the indoor experiment helps reducing the probability of mismatches given that the environment includes brick walls providing ambiguous texture information.

The outdoor sequence is composed of 3441 stereo pairs gathered in a public square of our home town (see fig. 6 top row). The full trajectory is approximately $140m$ long from the initial camera position. Figure 6 left column, shows the sequence of conditional independent local maps obtained with the technique described in section IV-A. Each map contains 100 features combining inverse depth and 3D points. The total number of maps built during the stereo sequence is 11. The result of D&C without applying the loop closing constraint is shown in fig. 6 middle column. As it can be observed, the precision of the map obtained is good enough to almost align the first and last submaps after all the trajectory has been traversed, even without applying loop closing constraints. Fig. 6 right column, presents the final result after closing the loop.

The second experiment was carried out inside one of our campus buildings in a walk of approximately $210m$ (see fig. 6 bottom row). The same process was run in order to obtain a full map from 8135 stereo pairs. This environment has a particular degree of difficulty due to ambiguous texture and the presence of extend zones of glass windows such as offices, corridors and cafeterias.

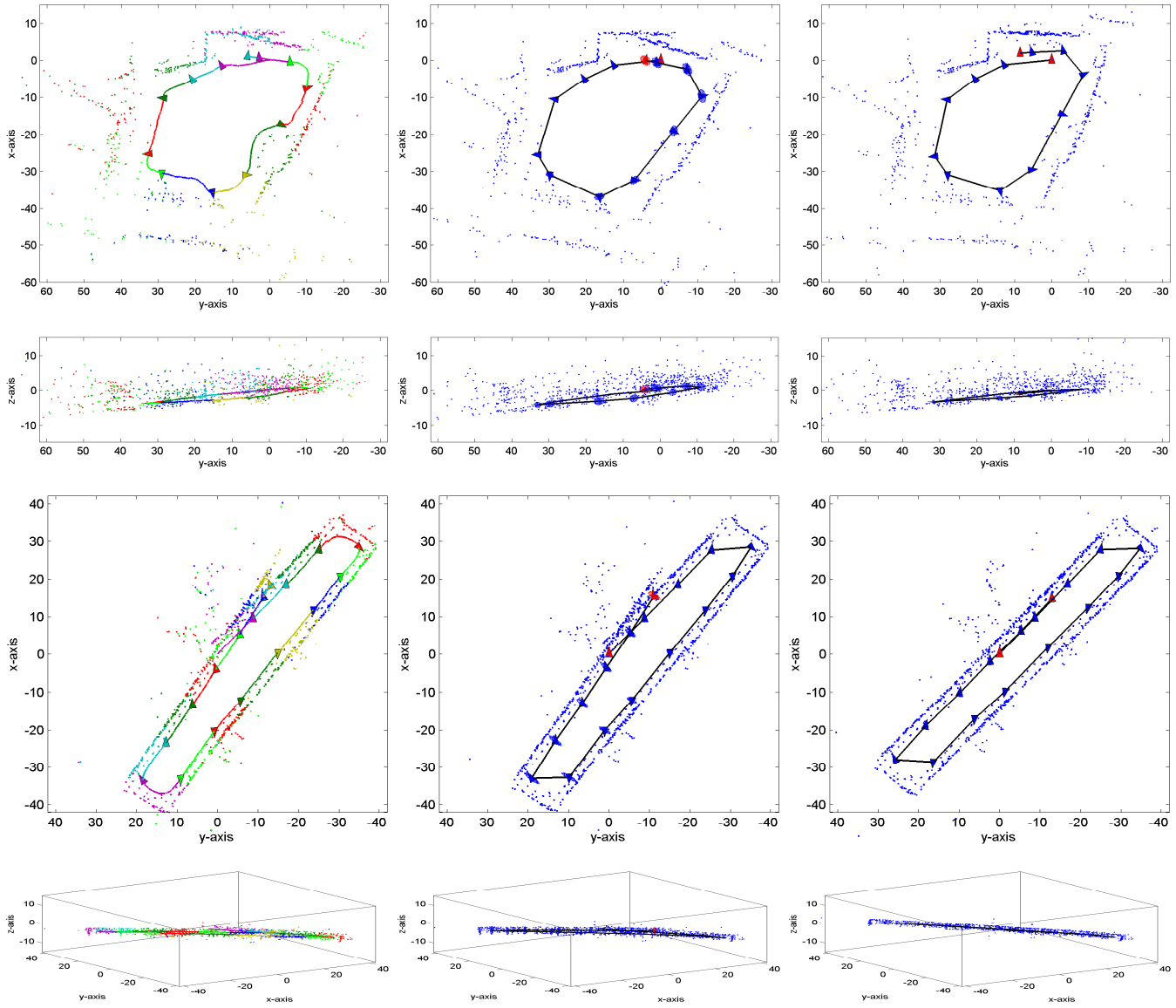

Fig. 6. *a.)* Outdoors experiment: 6DOF stereo SLAM on a public square. Both a XY projection (top row) and a YZ projection (top-middle row) are shown in order to illustrate the precision obtained.*b.)* Indoor experiment along a building environment: XY projection (bottom-middle row) and YZ projection (bottom row). The sequence of CI Local maps is represented with respect to the initial reference (left column); results obtained after running the D&C algorithm that joins and corrects the estimates (middle column); final map obtained when the loop closing constraint is imposed (right column). The scale factor and camera positions are well recovered thanks to the combined observations of 3D points and inverse depth points. The accompanying videos `VSLAM_video_outdoor.avi` and `VSLAM_video_indoor.avi` show the full execution of the outdoor and indoor experiments.

This can be noticed in the long distance points estimated in some of the maps, which are actually inside offices and the cafeteria (fig.6, left column). The result of CI D&C is shown in fig. 6 middle column, and the final result after loop closing is shown in fig.6 right column.

Our 6DOF SLAM system, even implemented in MATLAB, does not exceed 2 seconds per step, which is the worst case when building CI local maps. Fig. 7 shows how the running time system remains constant in most of the steps. Moreover, time peaks that appear when CI D&C takes place are below 8 seconds for the square experiment and 14 seconds for the indoor experiment, which are the maximum times required in the last step.

Using the Google Earth tool we can see that the map scale obtained and the trajectory followed by the camera is very close to the real scale. Fig. 9 illustrates comparative results. We loaded the MATLAB figure in Google Earth and set the scale parameter to the real scale. Given that we had no GPS nor compass measurements for the initial locations of the camera which are the base reference of each map, the position and orientation of the figure over the map were adjusted by hand. It can

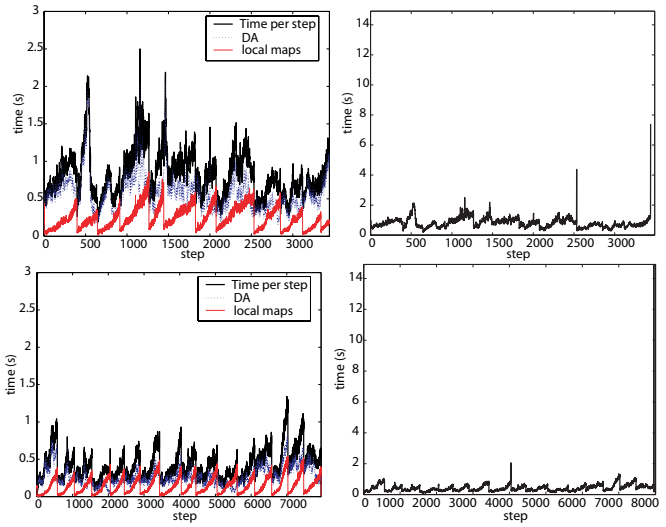

Fig. 7. Running time per step of all associated processes: a detailed analysis of the Features extraction, Local Mapping (labeled as local maps) and Data Association (DA) times (left); total time per step where the peaks represent the joins performed by the CI D&C algorithm (right). Outdoor environment: the Public square (top). Indoor environment (bottom).

be noticed that angles between the square sides and the shape of the walls of the surrounding environment have been captured with precision.

VI. Discussion

As presented in the introduction, several works have demonstrated successful Visual SLAM systems in small environments using monocular or stereo cameras. There are several important factors that limit the extension of these results to large scale environments.

First, the computational complexity and consistency of the underlying SLAM technique. In this work we have presented a novel algorithm that builds conditionally independent local maps in constant time and combines them in an optimal way in amortized linear time. Although the experiments presented here were processed in MATLAB, we expect that the extension to stereo of our current real-time implementation [18] will be able to build local maps up to 100 features in real time, with updates at 25Hz. The D&C map joining, loop detection and loop closing can be implemented on a separate thread, taking advantage of current multiple core processors.

In the case of monocular SLAM, another important limiting factor is the intrinsic unobservability of the scale. This problem can be addressed using additional sensors such as the vehicle odometry, GPS or inertial units. When they are not available, the scale can be initialized using some a priori knowledge about the environment such as the size of a known object visible

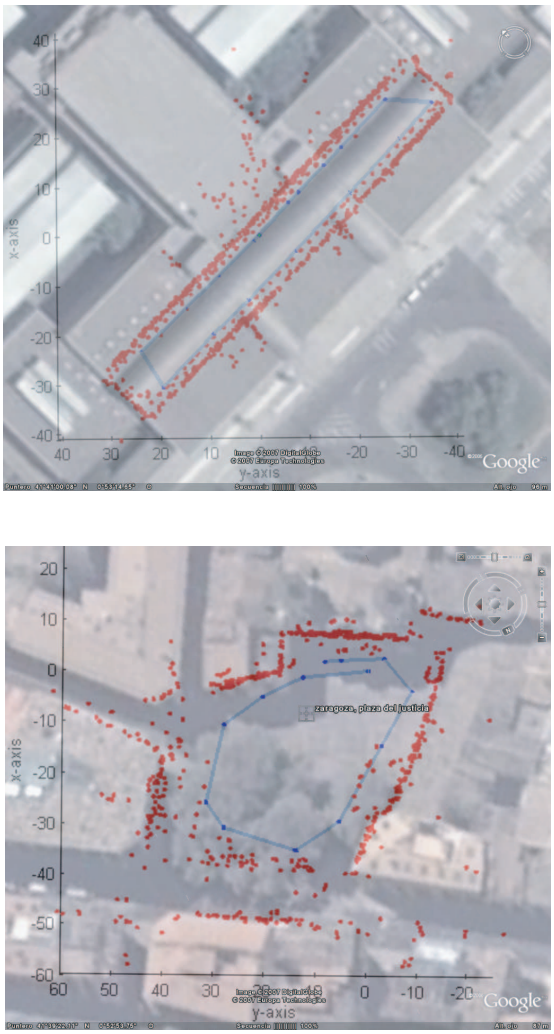

Fig. 9. Stereo visual SLAM recovers the true scale: the building environment (top) and the Public square (bottom) overlapping Google Earth.

at the start [15] or the initial speed of the camera [56]. However, in large environments, unless scale information is injected on the system periodically, the scale of the map can slowly drift (see for example, the experiments in [18]). Another critical issue appears when the scene is mostly planar and perpendicular to the optical axis. In this situation, with a monocular camera it is very difficult to distinguish between camera translation and rotation, unless a wide field of view (FOV) is used.

To illustrate these difficulties, we have processed our indoor and outdoor experiments using only the information from the right camera. As we are now using a bearing-only system, all the features are initialized using the inverse depth representation. To bootstrap the system, we have introduced a initial estimated speed for the camera of 1m/s. Apart from that, our Visual SLAM

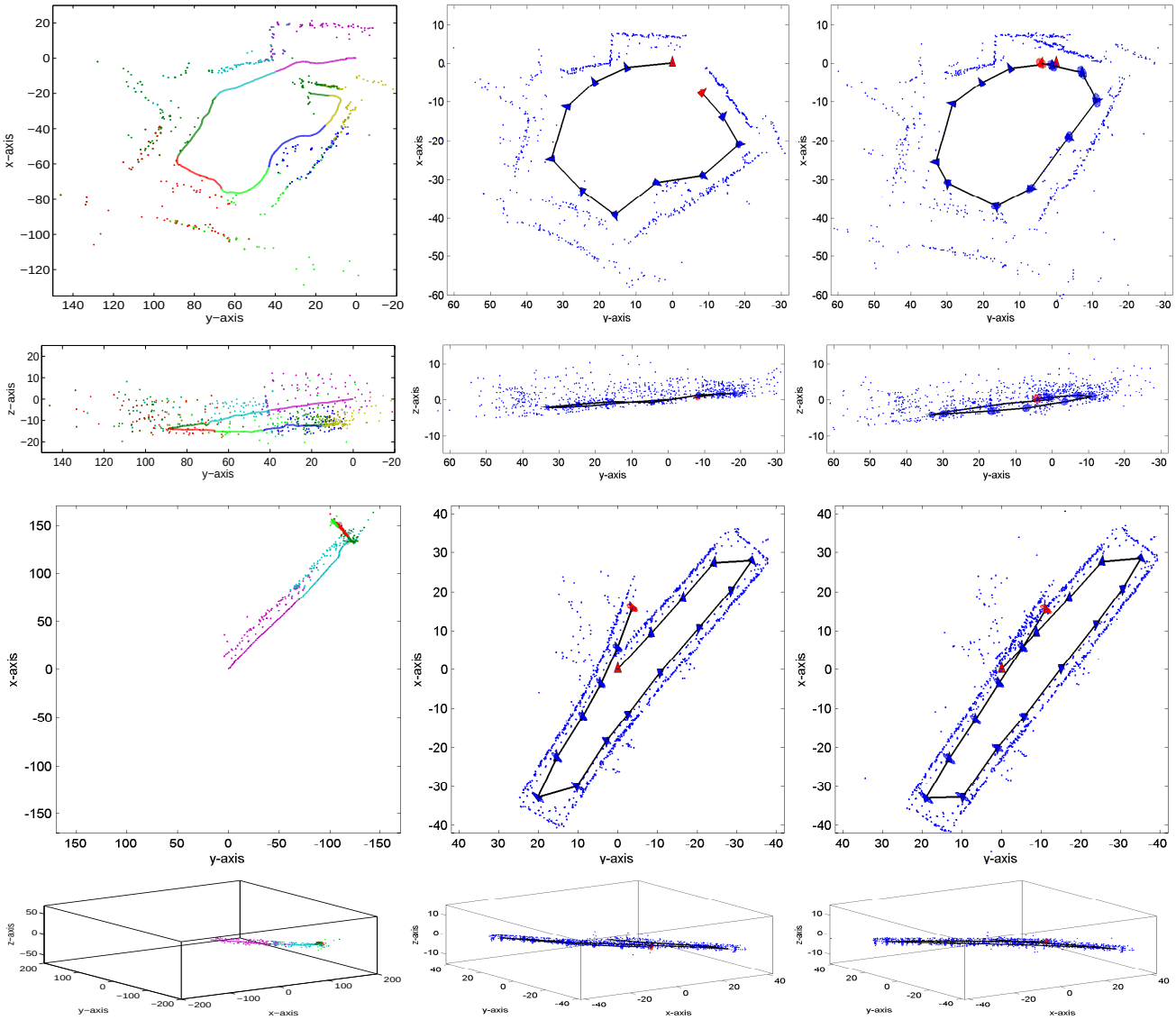

Fig. 8. Comparison of the outdoor and indoor maps obtained before the loop closure using three different techniques: monocular SLAM with inverse depth points (left), stereo SLAM with 3D points (middle) and the proposed stereo SLAM with 3D points and inverse depth points (right)

algorithm remains unchanged. The resulting maps are represented in the left column of figure 8. As it can be seen, the scale obtained by the system drifts (compare the beginning of the loop with the end). Also in the outdoor experiment, at a certain point, the system misinterprets the camera translation as a rotation, and the map gets corrupted. Here we are using a camera with FOV of 65 degrees. The results obtained in the same environment with a FOV of 90 degrees are significantly more robust [59]. In the indoor experiment with a monocular camera, as the objects are much closer to the camera, most of the features disappear fast from the FOV when the camera turns leading to a bad estimation of its position and consequently divergence in the map estimate.

We have also processed the sequences with our SLAM algorithm using conventional stereo, i.e. changed to initialize all the features whose disparity is bigger than one pixel as 3D points. Features without disparity are discarded because its depth cannot be computed by stereo. The immediate benefit is that the true environment scale is observable and the map corruption disappears (figure 8, middle column). However, for points that are more than 10m away from the camera, a Gaussian in xyz is a bad approximation for its true uncertainty. This is the reason for the map deformation that is clearly visible in the lower part of the outdoor experiment, where many features are at about 20m from the camera.

The proposed system (figure 8, right column) combines the advantages stereo and bearing only vision. On the one hand, the true scale is precisely obtained thanks

to the 3D information obtained by the stereo camera from close point features. On the other hand, the region with useful point features extends up to infinity, thanks to the inverse depth representation developed for bearing-only SLAM. The depth of the features that are far from the camera can be precisely recovered by the system if they are seen form viewpoints that are separated enough. In that case, they can be upgraded to 3D points for better efficiency [60]. Otherwise, they remain as inverse depth points and still provide very valuable orientation information that improves map precision and keeps the SLAM system stable when few close features are observed.

VII. Conclusions

In this paper we have shown that 6DOF visual mapping of large environments can be efficiently and accurately carried out using a stereo camera as the only sensor. One of the contributions of the paper is that information from features nearby and far from the cameras can be simultaneously incorporated to represent the 3D structure more precisely. Using close points provides scale information through the stereo baseline avoiding 'scale-drift', while inverse depth points are useful to obtain angular information from distant scene points.

Another contribution of the paper is the combination of two recent local mapping techniques to improve consistency and reduce complexity in the SLAM process. Using conditionally independent local maps, our system is able to properly share information related to the camera motion model, and common features between consecutive maps. Smoother transitions from map to map are achieved as well as better relative locations between local maps. By means of the simplicity and efficiency of the CI D&C SLAM algorithm, we can recover the full map very efficiently. The combination of both techniques adds robustness to the process without sacrificing precision.

In [51] we describe the performance of D&C SLAM when the vehicle carries out different types of trajectories. For some trajectories the cost of map joining can increase at some steps, depending of the size of the overlap between the maps to be joined: doing exploration the overlap is constant and the cost of map joining is small, when completing a loop traversal for a second time the overlap between the maps is total and the cost of joining will be much higher. Although we are able to close large indoor and outdoor loops, the algorithm used for loop closing strongly depends on detecting sets of features already stored in the map when the same area is revisited. It would be interesting to analyze other types of algorithms for loop closing, for instance the image to map algorithm proposed in [61].

Moreover, as we assume smooth motions, the relocation algorithm presented in [61] would enable the system to avoid failures in case of jitter.

There is also a restriction of the system to estimate pitch orientation due to the use of Euler angles. A combined solution using quaternions can mitigate the problem. This will be part of our future research.

Apart from upward looking cameras and jitter, there are no limitations to manoeuver the camera freely: it can be used in environments that include stairs and other terrain accidents. This kind of experiments makes part of the evaluation process for future work.

We will also focus on comparing our system with other stereo vision techniques such as visual odometry. We are very interested in studying the fusion of the stereo camera with other sensors like GPS or inertial systems in order to compare the precision obtained. We will consider other types of feature detectors as well, and their effect in the final result.

Acknowledgment

The authors would like to thank José M. M. Montiel and Javier Civera for the fruitful discussions.

References

[1] J. Folkesson and H. Christensen, "Graphical SLAM for outdoor applications," *Journal of Field Robotics*, vol. 23, no. 1, pp. 51–70, 2006.

[2] A. Nüchter, K. Lingemann, J. Hertzberg, and H. Surmann, "6D SLAM3D mapping outdoor environments: Research Articles," *Journal of Field Robotics*, vol. 24, no. 8-9, pp. 699–722, 2007.

[3] L. Ellekilde, "Dense 3D Map Construction for Indoor Search and Rescue," *Journal of Field Robotics*, vol. 24, no. 1-2, p. 71, 2007.

[4] J. Kim and S. Sukkarieh, "Airborne Simultaneous Localisation and Map Building," in *Proc. IEEE Int. Conf. on Robotics and Automation, (ICRA'03)*, 2003.

[5] M. Bryson and S. Sukkarieh, "Building a Robust implementation of Bearing-only Inertial SLAM for a UAV," *Journal of Field Robotics*, vol. 24, no. 1-2, pp. 113–143, 2007.

[6] M. Deans and M. Hebert, "Experimental Comparison of Techniques for Localization and Mapping Using a Bearing-Only Sensor," in *Int. Symp. on Experimental Robotics, (ISER'00). Lecture Notes in Control and Information Science*, S. S. D. Rus, Ed., vol. 271. Hawaii, USA: Springer, 2000, pp. 395–404.

[7] T. Fitzgibbons and E. Nebot, "Bearing Only SLAM using Colour-based Feature Tracking," in *2002 Australasian Conf. on Robotics and Automation*, Auckland, New Zealand, 2002.

[8] T. Bailey, "Constrained initialisation for bearing-only SLAM," in *Proc. IEEE Int. Conf. on Robotics and Automation, (ICRA'03)*, vol. 2, 2003.

[9] N. Kwok and G. Dissanayake, "An efficient multiple hypothesis filter for bearing-only SLAM," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, (IROS'04).*, vol. 1, 2004.

[10] N. Kwok, G. Dissanayake, and Q. Ha, "Bearing-only SLAM Using a SPRT Based Gaussian Sum Filter," in *Proc. of the IEEE Int. Conf. on Robotics and Automation, (ICRA'05).*, 2005, pp. 1109–1114.

[11] J. Sola, A. Monin, M. Devy, and T. Lemaire, "Undelayed initialization in bearing only SLAM," in *Proc. of the IEEE/RSJ Int. Con. on Intelligent Robots and Systems, (IROS'05)*, 2005, pp. 2499–2504.

[12] T. Lemaire, S. Lacroix, and J. Sola, "A practical 3D bearing-only SLAM algorithm," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, (IROS'05)*, 2005, pp. 2449–2454.

[13] P. Jensfelt, D. Kragic, J. Folkesson, and M. Bjorkman, "A framework for vision based bearing only 3D SLAM," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA'06)*, 2006.

[14] A. Gil, O. Reinoso, O. Martínez-Mozos, C. Stachniss, and W. Burgard, "Improving Data Association in Vision-based SLAM," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'06)*, Beijing, China, 2006.

[15] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "Monoslam: Real-time single camera slam," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1052–1067, June 2007.

[16] J. Folkesson, P. Jensfelt, and H. Christensen, "Vision SLAM in the measurement subspace," in *Proc. IEEE Int. Conf. on Robotics and Automation, (ICRA'05)*, 2005, pp. 30–35.

[17] P. Smith, I. Reid, and A. Davison, "Real-Time Monocular SLAM with Straight Lines," in *Proc. British Machine Vision Conference*, vol. 1, 2006, pp. 17–26.

[18] L. Clemente, A. J. Davison, I. D. Reid, J. Neira, and J. D. Tardós, "Mapping large loops with a single hand-held camera," in *Proc. Robotics: Science and Systems*, Atlanta, GA, USA, June 2007.

[19] T. Lemaire and S. Lacroix, "SLAM with panoramic vision," *Journal of Field Robotics*, vol. 24, no. 1-2, pp. 91–111, 2007.

[20] T. Goedemé, M. Nuttin, T. Tuytelaars, and L. Van Gool, "Omnidirectional Vision Based Topological Navigation," *Int. Journal of Computer Vision*, vol. 74, no. 3, pp. 219–236, 2007.

[21] E. Royer, M. Lhuillier, M. Dhome, and J. Lavest, "Monocular Vision for Mobile Robot Localization and Autonomous Navigation," *Int. Journal of Computer Vision*, vol. 74, no. 3, pp. 237–260, 2007.

[22] A. Davison, "Mobile Robot Navigation using Active Vision," Ph.D. dissertation, University of Oxford, 1998.

[23] A. J. Davison and D. W. Murray, "Simultaneous localization and map-building using active vision," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 865–880, 2002.

[24] A. Davison and N. Kita, "3D Simultaneous Localisation and Map-building using active vision for a robot moving on undulating terrain," in *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition, (CVPR'01)*, 2001.

[25] L. Iocchi, K. Konolige, and M. Bajracharya, "Visually realistic mapping of a planar environment with stereo," in *Proc. International Symposium on Experimental Robotics, (ISER'00)*, 2000.

[26] S. Se, D. Lowe, and J. Little, "Mobile Robot Localization and Mapping with Uncertainty using Scale-Invariant Visual Landmarks," *Int. J. Robotics Research*, vol. 21, no. 8, pp. 735–758, 2002.

[27] I. Jung and S. Lacroix, "High resolution terrain mapping using low altitude aerial stereo imagery," in *Proc. of the 9th Int. Conf. on Computer Vision*, 2003, pp. 946–951.

[28] E. Hygounenc, I. Jung, P. Soueres, and S. Lacroix, "The autonomous blimp project of LAAS-CNRS: Achievements in flight control and terrain mapping." *Int. Journal of Robotics Research*, vol. 23, no. 4, pp. 473–511, 2004.

[29] J. Saez, F. Escolano, and A. Penalver, "First Steps towards Stereo-based 6DOF SLAM for the Visually Impaired," in *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR'05)-Workshops-Volume 03*. IEEE Computer Society Washington, DC, USA, 2005.

[30] R. Sim, P. Elinas, M. Griffin, and J. Little, "Vision-based SLAM using the rao-blackwellised particle filter," in *IJCAI Workshop on Reasoning with Uncertainty in Robotics (RUR)*, 2005.

[31] R. Sim, P. Elinas, and J. Little, "A Study of the Rao-Blackwellised Particle Filter for Efficient and Accurate Vision-Based SLAM," *Int. Journal of Computer Vision*, vol. 74, no. 3, pp. 303–318, 2007.

[32] T. Lemaire, C. Berger, I. Jung, and S. Lacroix, "Vision-Based SLAM: Stereo and Monocular Approaches," *Int. Journal of Computer Vision*, vol. 74, no. 3, pp. 343–364, 2007.

[33] J. Sola, A. Monin, and M. Devy, "BiCamSLAM: Two times mono is more than stereo," in *Proc. IEEE Int. Conf. on Robotics and Automation, (ICRA'07)*, Rome, 2007.

[34] Z. Zhang and O.Faugueras, "Three-dimensional motion computation and object segmentation in a long sequence of stereo frames," *Int. Journal of Computer Vision*, vol. 7, no. 3, pp. 211–241, 1992.

[35] N. Simond and P. Rives, "Trajectography of an uncalibrated stereo rig in urban environments," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, (IROS'04).*, vol. 4, 2004.

[36] D. Nister, O. Naroditsky, and J. Bergen, "Visual odometry for ground vehicle applications," *Journal of Field Robotics*, vol. 23, no. 1, pp. 3–20, 2006.

[37] A. Comport, E. Malis, and P. Rives, "Accurate Quadri-focal Tracking for Robust 3D Visual Odometry," in *Proc. IEEE Int. Conf. on Robotics and Automation, (ICRA'07)*, April, 2007.

[38] M. Maimone, Y. Cheng, and L. Matthies, "Two Years of Visual Odometry on the Mars Exploration Rovers," *Journal of Field Robotics*, 2007.

[39] M. Agrawal and K. Konolige, "Real-time Localization in Outdoor Environments using Stereo Vision and Inexpensive GPS," in *Int. Conf. on Pattern Recognition (ICPR'06)*, 2006.

[40] K. Konolige, M. Agrawal, R. Bolles, C. Cowan, M. Fischler, and B. Gerkey, "Outdoor mapping and navigation using stereo vision," in *Intl. Symp. on Experimental Robotics*, 2006.

[41] J. A. Castellanos, J. Neira, and J. D. Tardós, "Limits to the consistency of EKF-based SLAM," in *5th IFAC Symposium on Intelligent Autonomous Vehicles*, Lisbon, Portugal, 2004.

[42] J. Castellanos, R. Martinez-Cantin, J. Tardós, and J. Neira, "Robocentric Map Joining: Improving the consistency of EKF-SLAM," *Robotics and Autonomous Systems*, vol. 55, no. 1, pp. 21–29, January 2007.

[43] S. Thrun, Y. Liu, D. Koller, A. Y. Ng, Z. Ghahramani, and H. Durrant-Whyte, "Simultaneous Localization and Mapping with Sparse Extended Information Filters," *Int. J. Robotics Research*, vol. 23, no. 7-8, pp. 693–716, 2004.

[44] R. Eustice, M. Walter, and J. Leonard, "Sparse extended information filters: Insights into sparsification," in *IEEE Int. Workshop on Intelligent Robots and Systems*, Edmonton, Alberta, Canada, August 2005.

[45] U. Frese, *Treemap: An o(logn) algorithm for simultaneous localization and mapping*. Springer Verlag, 2005, ch. Spatial Cognition IV, p. 455476.

[46] ——, "Efficient 6DOF SLAM with Treemap as a Generic Backend," in *IEEE Int. Conf. on Robotics and Automation (ICRA'07)*, 2007.

[47] F. Dellaert and M. Kaess, "Square root SAM: Simultaneous localization and mapping via square root information smoothing," *Int. J. Robotics Research*, vol. 25, no. 12, December 2006.

[48] M. Kaess, A. Ranganathan, and F. Dellaert, "iSAM: Fast Incremental Smoothing and Mapping with Efficient Data Association," in *Proc. IEEE Int. Conf. on Robotics and Automation, (ICRA'07)*, 2007.

[49] J. D. Tardós, J. Neira, P. M. Newman, and J. J. Leonard, "Robust mapping and localization in indoor environments using sonar

data," *Int. J. Robotics Research*, vol. 21, no. 4, pp. 311–330, 2002.

[50] L. M. Paz, P. Jensfelt, J. D. Tardós, and J. Neira, "EKF SLAM updates in *O(n)* with Divide and Conquer SLAM," in *Proc. IEEE Int. Conf. Robotics and Automation*, Rome, Italy, April 2007.

[51] L. M. Paz, J.Guivant, J. D. Tardós, and J. Neira, "Data association in *O(n)* for divide and conquer SLAM," in *Proc. Robotics: Science and Systems*, Atlanta, GA, USA, June 2007.

[52] J. M. M. Montiel, J. Civera, and A. J. Davison, "Unified inverse depth parametrization for monocular SLAM," in *Proc. Robotics: Science and Systems*, Philadelphia, USA, August 2006.

[53] A. J. Davison, "Real-time simultaneous localisation and mapping with a single camera," in *Proc. Int. Conf. Computer Vision*, Nice, Oct 2003.

[54] L. M. Paz, P. Piniés, J. D. Tardós, and J. Neira, "Measurement Equation for Inverse Depth points and Depth points," Internal Report RR-08-06, Dept. Informática e Ingeniería de Sistemas, Universidad de Zaragoza, Spain, 2008.

[55] J. Civera, A. Davison, and J. Montiel, "Inverse Depth to Depth Conversion for Monocular SLAM," in *Proc. IEEE Int. Conf. Robotics and Automation*, Roma, Italy, April 2007.

[56] P. Pinies and J. D. Tardós, "Scalable SLAM building Conditionally Independent Local Maps," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, (IROS'07)*, November, 2007.

[57] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[58] J. Neira and J. D. Tardós, "Data association in stochastic mapping using the joint compatibility test," *IEEE Trans. Robotics and Automation*, vol. 17, no. 6, pp. 890–897, 2001.

[59] P. Pinies and J. D. Tardós, "Linear time slam building conditionally independent local maps: Application to monocular vision," 2008, to appear.

[60] J. M. M. Montiel, J. Civera, and A. J. Davison, "Unified inverse depth parametrization for monocular SLAM," *IEEE Trans. Robotics*, 2008, to appear.

[61] B. Williams, P. Smith, and I. Reid, "Automatic relocalisation for a single-camera simultaneous localisation and mapping system," in *Proc. IEEE Int. Conf. Robotics and Automation*, Roma, Italy, April 2007, pp. 2784–2790.