3D Map Registration using Vision/Laser and Inertial Sensing

Luiz G. B. Mirisola, Jorge Lobo and Jorge Dias
ISR - Institute of Systems and Robotics
University of Coimbra - Portugal
{lgm,jlobo,jorge}@isr.uc.pt

***Abstract*— A point cloud registration method is proposed in this article, and experimental results are presented for long three-dimensional map sequences obtained from a moving observer.**

In vision based systems used in mobile robotics the perception of self-motion and the structure of the environment is essential. Inertial and earth field magnetic pose sensors can provide valuable data about camera ego-motion, as well as absolute references for the orientation of scene structure and features. In this work we explore the fusion of inertial and magnetic sensor data with range sensing devices. Orientation measurements from the inertial system are used to rotate the obtained 3D maps into a common orientation, compensating the rotational movement. Then, image correspondences are used to find the remaining translation. Results are presented using both a stereo camera and a laser range finder as the ranging device. The laser range finder also needs a single camera to stabilish pixel correspondence.

The article overviews the camera-inertial and camera-laser calibration processes used. The map registration approach is presented and validated with experimental results on indoor and outdoor environments.

***Index Terms*— Inertial Sensors, 3D Mapping, Laser Sensors**

I. Introduction

Vision or 3D imaging systems in robotic applications can be rigidly coupled with an Inertial Measurement Units (IMUs) and magnetic sensors, which complement it with sensors providing direct measures of orientation relative to the world north-east-up frame, such as magnetometers (that measure the earth magnetic field) and accelerometers (that measure gravity) [13]. Micromachining enabled the development of low-cost single-chip inertial and magnetic sensors that can be easily incorporated together alongside the camera and other sensors such as a laser range finder (LRF).

Calibration techniques find the rigid body rotation between the camera and IMU frames [9, 10], and between the camera and LRF frames [17]. Then, the orientation of the camera or LRF in the world can be calculated from the IMU orientation measurement. The knowledge of the range sensing device orientation should allow faster processing or the usage of simpler motion models in registration tasks. For example, these two sensory modalities can be explored to improve robustness on image segmentation and 3D structure recovery from images [8, 15] or independent motion segmentation [11].

Figure 1 shows a camera, an IMU, and an LRF mounted on a pan-tilt.

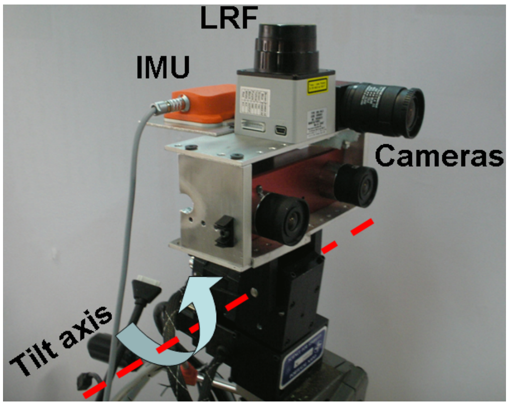


Fig. 1. The sensors utilized for 3D mapping.

3D mapping with color images and a rotating LRF was already performed [16], but without any calibration process to calibrate the rotation between the sensor frames, and using only ICP to register the point clouds. Rotating LRFs were also used to recover 3D range scans, that were matched to build maps, in room scale with small mobile robots [6], or in larger scale in urban mapping with cars as the sensor platform [18].

To detect outliers on stereo registration two geometric constraints were proposed in [5]. The single constraint on translation vectors proposed here subsumes the other two.

Our first aim is to register 3D point clouds obtained by a mobile observer carrying a stereo camera [15]. Correlation based stereo depth maps were obtained for each frame. Having the camera-inertial system calibrated, the camera orientation in the world was calculated from the IMU orientation measurements, and the point clouds were rotated to a common levelled and earth referenced frame. Then the remaining 3D translation to register the successive point clouds was estimated by tracking image targets over successive frames, and subtracting their 3D position. Fully registered point clouds can therefore be obtained.

Then, on this paper the same technique is applied to a different setup with a LRF as the ranging sensing device instead of a stereo system. Images from a single camera will still be used to track targets and relate them with the 3D points, after a LRF-camera calibration process. The IMU will still be used to rotate the point clouds to a leveled reference frame, as the rotation LRF-IMU can be calculated from the result of both calibration processes.

Although accumulation of errors does not allow the registration of a long sequence of point clouds by registering only pairs of point clouds taken from adjacent frames, it is possible to register and combine into a larger, aggregated point cloud a limited sequence of neighbouring point clouds around one taken as reference.

The next subsections define the reference frames utilized, reviews the calibration processes, and present the experimental platform. Section II describes our present approach, followed by experimental results on section III and finally the conclusions on section IV.

A. Definitions of reference frames

In the scenario of figure 2(a), the IMU is mounted with a stereo camera. In the scenario of figure 2(b), an inertial system is rigidly mounted with a LRF and a single calibrated camera. Hence the following reference frames are defined:

- **Camera Frame** $\{\mathcal{C}\}$: A common pinhole camera projection model. The origin is placed at the *camera center*, the axis z is the *depth* from the camera, and the axes x and y form the *image plane*. In the stereo scenario, the $\{\mathcal{C}\}$ frame is defined by the left camera.
- **Inertial Frame** $\{\mathcal{I}\}$: The inertial orientation output is the rotation between the $\{\mathcal{I}\}$ and the $\{\mathcal{W}\}$ frames.
- **World Frame** $\{\mathcal{W}\}$: A Latitude Longitude Altitude (LLA) frame.
- **Laser Frame** $\{\mathcal{L}\}$ Its origin is the convergence point of the laser beams. Its axes are alligned with the laser beams as shown in figure 2.
- **Rotated Device Frame** $\{\mathcal{R}\}$: This frame shares its origin with the $\{\mathcal{C}\}$ (stereo scenario) or $\{\mathcal{L}\}$ (LRF scenario) frames, but its axes are aligned with the world frame $\{\mathcal{W}\}$ (see figure 5).

B. Calibration of fixed rotations

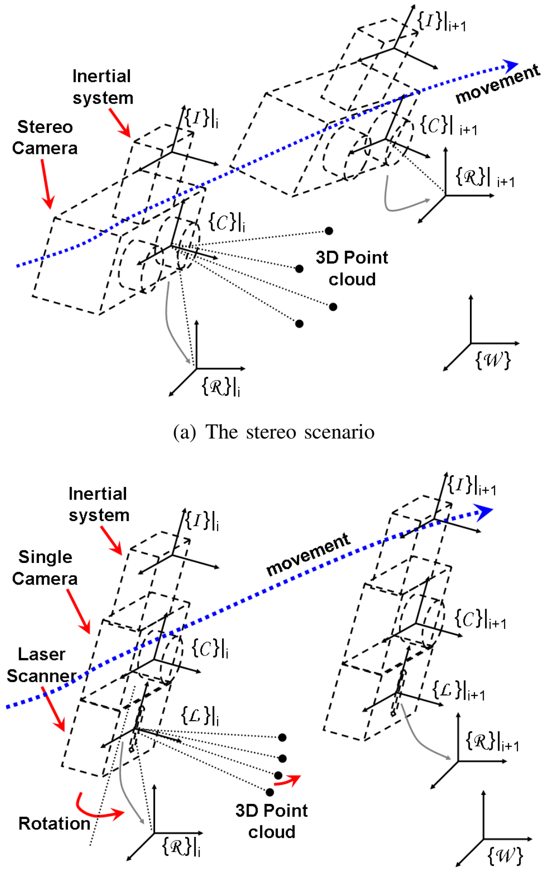
The camera-inertial calibration [9, 10] outputs the constant rotation ${}^{\mathcal{I}}R_{\mathcal{C}}$ between the camera $\{\mathcal{C}\}$ and inertial $\{\mathcal{I}\}$ frames. It is implemented as a Matlab toolbox [12]. Two examples of calibration images are shown in figure 3, where a chessboard was placed in the vertical position, so that its vertical lines provide an image-based measurement of the gravity direction to be registered with the gravity measurements provided by the accelerometers.

On this paper the fixed rotation ${}^{\mathcal{C}}R_{\mathcal{L}}$ between the laser $\{\mathcal{L}\}$ and camera $\{\mathcal{C}\}$ frames was found by reprojecting the image pixels onto the point cloud and adjusting the rotation until 3D structures are correctly painted.

The fixed rotation ${}^{\mathcal{I}}R_{\mathcal{L}}$ between the LRF and IMU frames can be calculated as ${}^{\mathcal{I}}R_{\mathcal{L}} = {}^{\mathcal{I}}R_{\mathcal{C}} \cdot {}^{\mathcal{C}}R_{\mathcal{L}}$.

C. Experimental Platform

The LRF is a Hokayo URG-04LX (figure 1), used together with a single camera Allied Guppy-36C[1], calibrated with the Camera Calibration toolbox[3]. The stereo camera is a Videre STH-DCSG-C Stereo Head [19]. To calibrate the cameras and compute range from stereo images we use the Small Vision System (SVS) Software [7]. All sensors are rigidly mounted together with the inertial and magnetic sensor package MTi from Xsens [20]



(a) The stereo scenario

(b) The laser scanner scenario

Fig. 2. Moving observer and world fixed frames of reference

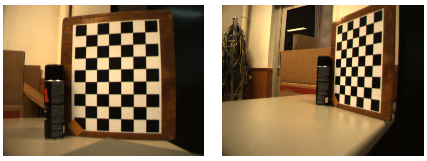


Fig. 3. Two examples of calibration images used to calibrate the camera-inertial system.

II. Registering 3D Point Clouds

Figure 4 shows the data flow for point cloud registration. On the left the inputs are shown: for frame i, the point cloud, the camera image, and the inertial orientation measurement. Two other inputs are constant for all frames: the reference image, and the fixed rotation matrix between the IMU and the ranging device (${}^{I}R_{L}$ or ${}^{I}R_{C}$).

A. Obtaining images and point clouds.

For each time index i, the camera provides intensity images $I_i(u, v)$ where u and v are pixel coordinates. The laser scanner outputs a set of 3D points ${}^{\mathcal{L}}\mathbb{P}|_i$, expressed in the laser frame of reference $\{\mathcal{L}\}$. The 3D positions of each point are given by a laser projection model as defined in [17]. For each point the pantilt position is defined by the tilt angle (φ), as the pan axis was not moved, and the LRF supplies a distance measurement ρ and a beam angle θ in the scan plane.

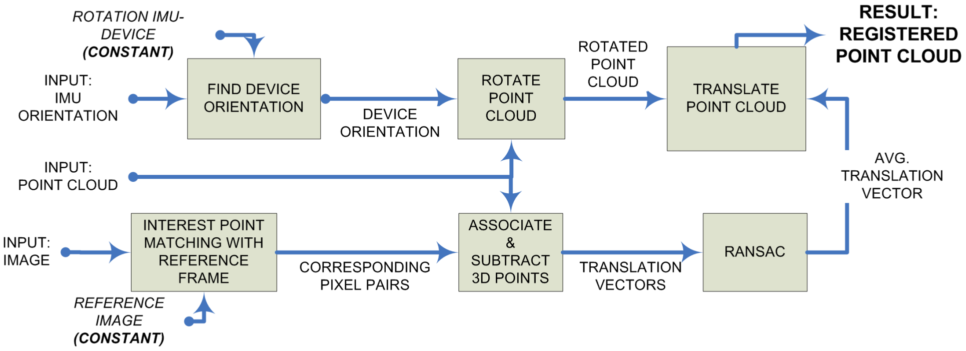


Fig. 4. The data flow for the registration of each 3D point cloud.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} c_\varphi c_\theta & -c_\varphi s_\theta & s_\varphi & c_\varphi d_x + s_\varphi d_z \\ s_\theta & c_\theta & 0 & 0 \\ -s_\varphi c_\theta & s_\varphi s_\theta & c_\varphi & -s_\varphi d_x + c_\varphi d_z \end{bmatrix} \begin{bmatrix} \rho \\ 0 \\ 0 \\ 1 \end{bmatrix} \tag{1}$$

where c and s represent the cosine and sine functions and the calculated x, y, z coordinates are given in the $\{\mathcal{L}\}$ frame.

Each 3D point is be reprojected into the camera frame, and if it falls in the camera field of view, it is associated with an image pixel (u, v) on the image I_i, with a corresponding intensity gray level $c = I_i(u, v)$.

When using stereo cameras, the stereo image pair is processed to calculate a depth for most pixels, yielding a set of 3D points ${}^{\mathcal{C}}\mathbb{P}|_i$ directly in the $\{\mathcal{C}\}$ frame. In both cases each point in the set retains both 3D position and gray level.

$$P(x, y, z, c) \in {}^{\{\mathcal{C},\mathcal{L}\}}\mathbb{P}|_i$$

B. Rotate to Local Vertical and Magnetic North

The measured inertial orientation for the time index i, expressed as a rotation matrix ${}^{\mathcal{W}}R_{\mathcal{I}}|_i$, rotates the inertial frame $\{\mathcal{I}\}|_i$ into the world frame $\{\mathcal{W}\}$.

If stereo cameras are used as the ranging device, the point cloud ${}^{\mathcal{C}}\mathbb{P}|_i$ are rotated by the rotation ${}^{\mathcal{W}}R_{\mathcal{C}}|_i = {}^{\mathcal{W}}R_{\mathcal{I}}|_i \cdot {}^{\mathcal{I}}R_{\mathcal{C}}$, that rotates the camera frame into the world frame. As figure 5 shows for two point clouds, the purpose of this rotation is to align all point clouds to the earth-referenced $\{\mathcal{R}\}$ frame, i.e., North, East and vertical directions, as indicated by the inertial and magnetic orientation measurements.

For the LRF, a similar step is taken, but it is necessary also to compensate for the pantilt position at the moment the inertial measurement was taken. The pantilt rotation is represented by $R(\varphi, \psi)$, the rotation matrix equivalent to the Euler angles φ, ψ, 0 considered at the relevant axes. For each time index $i = 1 \ldots n$, we define the matrix ${}^{\mathcal{W}}R_{\mathcal{L}}|_i = {}^{\mathcal{W}}R_{\mathcal{I}}|_i \cdot {}^{\mathcal{I}}R_{\mathcal{L}} \cdot R(\varphi, \psi)$ as the rotation that brings a point from the laser frame $\{\mathcal{L}\}|_i$ into the world frame $\{\mathcal{W}\}$, and apply this rotation to 3D points on the point cloud ${}^{\mathcal{L}}\mathbb{P}|_i$, generating a point cloud ${}^{\mathcal{R}}\mathbb{P}|_i$ in the rotated frame of reference. The fixed rotation ${}^{\mathcal{I}}R_{\mathcal{L}}$ was obtained by verifying as shown in section I-B.

After this step only a translation is missing to register the point cloud into the $\{\mathcal{W}\}$ frame.

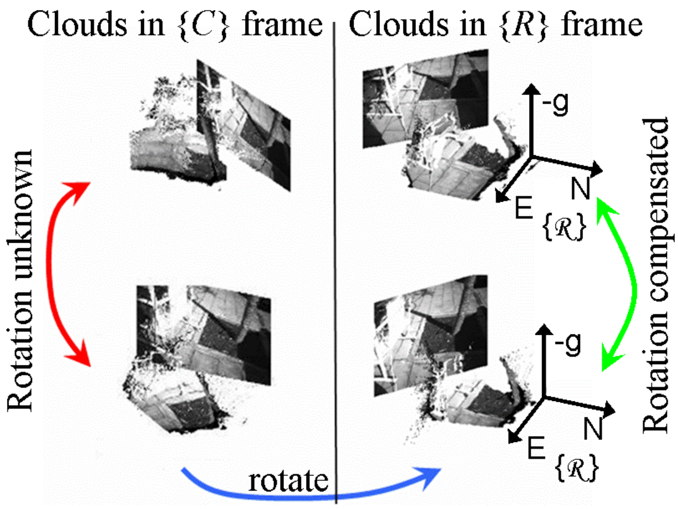


Fig. 5. Compensating the rotation: point clouds aligned on inertial levelled frame.

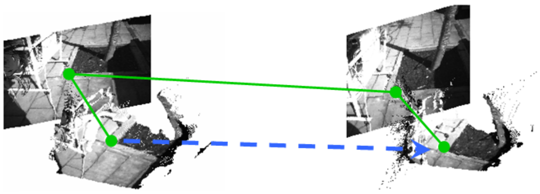


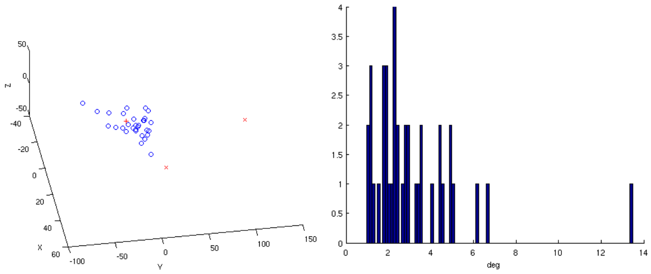
Fig. 6. 2D image matching and corresponding pair of 3D points with a translation vector.

C. Translation from Image Correspondences

The translation component between the point clouds ${}^{\mathcal{R}}\mathbb{P}|_i$ and ${}^{\mathcal{R}}\mathbb{P}|_{i+1}$ can be obtained by tracking fixed targets in the images I_i and I_j of the scene. The tracked image features must have the corresponding 3D points $\mathbf{P}|_i \in {}^{\mathcal{R}}\mathbb{P}|_i$ and $\mathbf{P}|_j \in {}^{\mathcal{R}}\mathbb{P}|_j$ in each point cloud, such that each tracked feature yields a translation vector in the form:

$$\Delta \overrightarrow{\mathbf{t}} = \mathbf{P}|_i - \mathbf{P}|_j$$

Figure 6 shows two corresponding 3D points in two overlapping point clouds, obtained from a corresponding pixel pair on the respective images. The dashed arrow is the translation vector, with the corresponding 3D points and image pixels connected by full lines.



(a) Difference between translations vectors and the mean vector (in mm).

(b) Histogram of the angle between the translation vectors and the mean vector.

Fig. 7. An example of the usage of *RANSAC* to detect *outliers*.

Interesting points are found by the SURF algorithm [2]. Assuming that the majority of interesting points are from the static background, random sample consensus (RANSAC [4]) is used to reject outliers. Interesting points wrongly matched and 3D points wrongly positioned due to errors on the stereo disparity image are detected as outliers by the same RANSAC procedure.

As the model used on RANSAC is very simple, involving averaging and subtracting 3D vectors, the RANSAC procedure runs very fast. Since there is not an absolute pose reference, one reference frame $\{\mathcal{R}\}|_0$ is arbitrarily choosen as the global frame where the other point clouds will be registered to.

1) The stereo camera case: When point clouds are obtained from stereo cameras, dense stereo algorithms can recover disparities and thus 3D points for many (or most) image pixels. Then the association of image pixels with 3D points is trivial, and, for most matched pairs of interest points, there exist a corresponding 3D point pair. The ones for which there is not an associated 3D point pair can be simply discarded.

Both mismatched interesting points and wrong stereo disparities are detected as outliers by the same RANSAC procedure. Figure 7(a) is a plot of the set of translation vectors for one image pair. The plotted circles are the differences between each inlier vector and their mean (indicated by $'+'$) - i.e., if all vectors were equal, all circles would appear on the origin. The 'x's are outliers, which were detected and eliminated. Figure 7(b) is a histogram of the angle between the translation vectors and the mean vector - most point approximatelly to the same direction, except a few outliers (the crosses on the left graph).

2) The LRF case: In the case of the point clouds from the LRF, as its angular resolution is tipically less than the image angular resolution, most pixels do not have an associated 3D point. When one pixel $\mathbf{x}$, belonging to a matched interesting point pair, do not have an associated 3D point, the closest pixel $\mathbf{x}^*$ with an associated 3D point in its neighborhood is found. If the image distance $|\mathbf{x} - \mathbf{x}^*|$ is less than a small threshold (2 pixels in our experiments) that 3D point substitutes the missing one. This approximation increases the measurement error for the translation. Therefore, as suggested in [14], for each corresponding pixel pair $(\mathbf{x}_i, \mathbf{x}_j)$ in the images I_i and I_j, the value $w = (|\mathbf{x}_i - \mathbf{x}_i^*| + |\mathbf{x}_j - \mathbf{x}_j^*|)^{-1}$ is defined as a weight in the averaging of the resulting translation vector T, that become, for N corresponding pixel pairs:

$$T = (\sum_{k=1}^{N} {}_{k=1}^{N} w_k)^{-1} (\sum_{k=1}^{N} w_k \Delta \overrightarrow{\mathbf{t}}_k) \qquad (2)$$

D. A non-iterative alternative to RANSAC.

To detect outliers, instead of using RANSAC, geometric constraints can be exploited, avoiding iterative techniques. An example are the two constraints proposed by [5].

Consider two pairs of corresponding 3D points on the i and j frames, $(\mathbf{P}|_i, \mathbf{P}|_j)$ and $(\mathbf{Q}|_i, \mathbf{Q}|_j)$. The first constraint concerns the invariance of the length of the $\mathbf{P}|_i - \mathbf{Q}|_i$ vector under a rigid transformation.

The second constraint limits the angle between a vector formed by two 3D points before and after the motion. With the orientation measured, it can be more tightly enforced with a statistically significant boundary if the error on orientation measurements is known.

As these constraint will never be exactly satisfied in practice, the problem consists in determining if the difference verified is consistent with the expected errors on the process; if it is not, one of the points must be an outlier. Therefore these constraints are checked against all possible pairs of corresponding 3D points, and the maximum subset of consistent measurements is selected.

Altough this method is not interactive, it requires significantly more time than the RANSAC method outlined on section II-C. Also, the difference between translation vectors is a single constraint that is not satisfied if any of the two other constraints is not satisfied, and therefore it subsumes both. Therefore the method of section II-C was chosen to be used on the experiments of this paper, and the non-iterative method was abandoned.

E. Filtering out redundant points

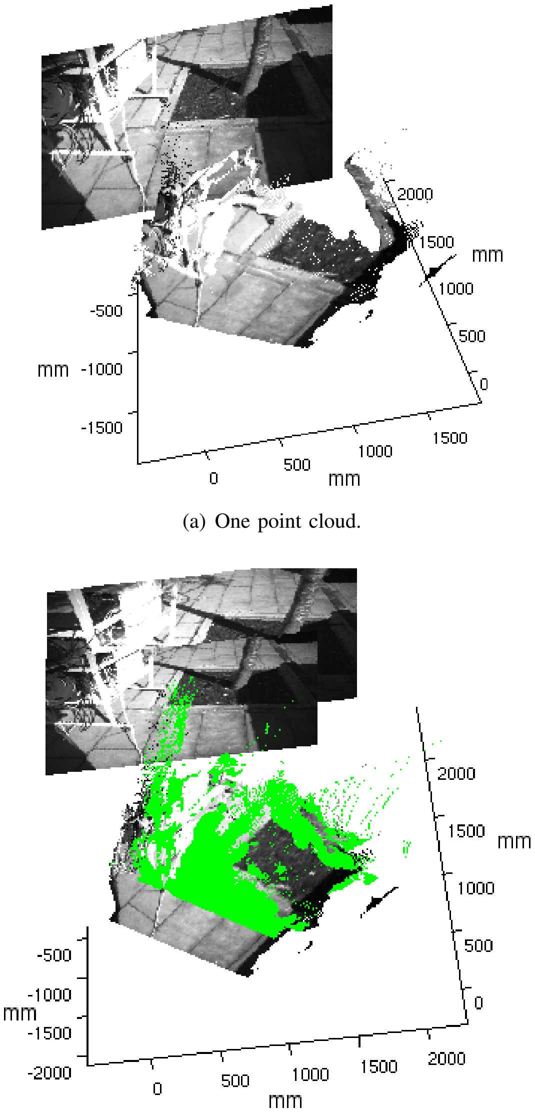
The point clouds being registered have large overlap and many redundant points. To save memory, new points too close to a point already present on the cloud should be rejected. But, as the number of points is large, it is too slow to check linearly all the stored points to test if a new point is redundant.

Additionally, the point clouds should be filtered, eliminating points wrongly positioned due to range sensing errors. Isolated points must be deleted to generate a smoother point cloud.

One approach would be to divide the covered space in voxels and mark each voxel as occupied or free. The disavantage of this approach is that the number of voxels increases with the covered space, and many voxels are empty. This waste of memory should be avoided to be able to cover a larger space.

Another well-known approach, which has been implemented here, keeps only a 3D point cloud and a hash table indexing all points by their coordinates. When a new point is inserted, the hash table retrieves a list of potentially close points, rejecting the new point if it is redundant.

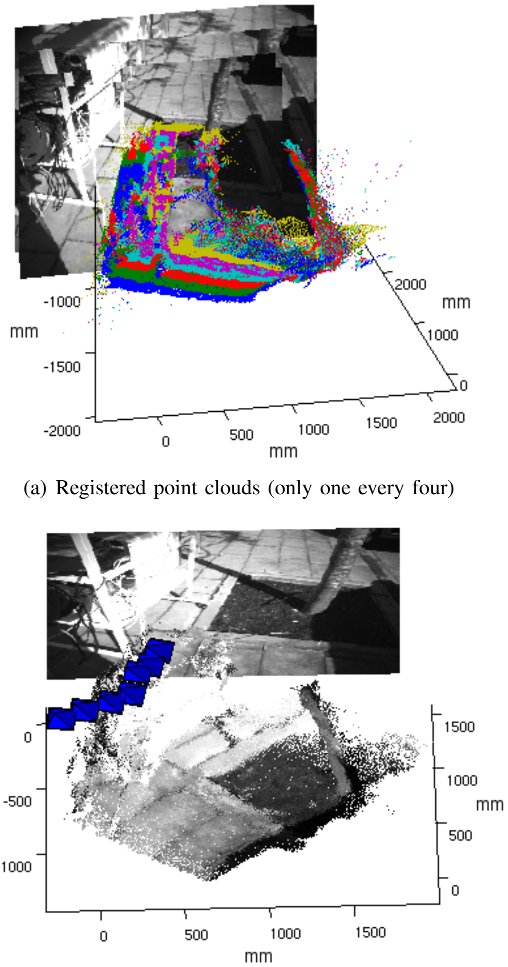
Doubious points are eliminated by deleting points that were not seen in a sufficient number of frames. To keep track of this, every point is associated to a counter, that is incremented every time there is an attempt to insert a new point on the same position. Each counter can be incremented only once per frame. In such a way the frames “vote” for each point.



(a) One point cloud.

(b) Two registered point clouds.

Fig. 8. Point clouds from the sidewalk dataset.



(a) Registered point clouds (only one every four)

(b) The resulting, filtered point cloud. The camera poses are shown as blue pyramids.

Fig. 9. The result of registering point clouds for 27 successive frames.

III. Experimental Results

A. *Using stereo cameras*

Figure 8(a) shows one point cloud from an outdoor image sequence, with its left camera image displayed on the back. Figure 8(b) shows two registered point clouds, one in green, the other in the original gray level color, with their respective images displayed behind them.

Figure 9 shows, on the left, a set of registered point clouds of the same sidewalk sequence, and on the right, the resulting point cloud after registering together a sequence of 27 successive point clouds, and filtering out points imaged in less than 4 frames. In the left figure, only one every four point clouds is shown, to ease visualization.

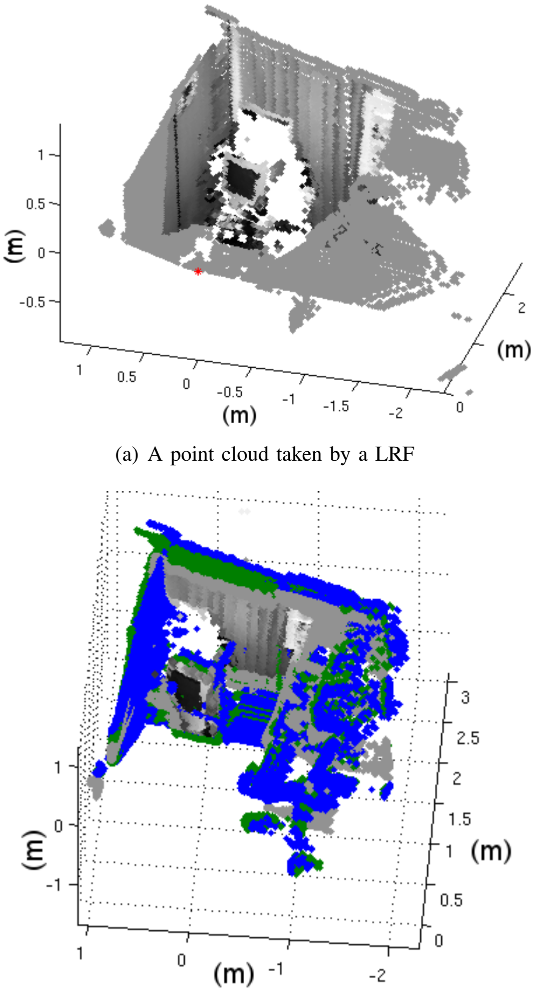
The pyramids (one for every four camera poses) on figure 9(b) represent the camera trajectory and orientation (the cameras point towards the pyramids base). The *RANSAC* threshold for membership in the inlier set was $5\,cm$. The minimum acceptable number of inliers was 20. For each reference frame, between 20 and 50 frames were registered, representing between $1.5\,s$ and $3\,s$ worth of data at $15\,fps$.

B. *Using a LRF*

To generate a 3D point cloud, the LRF was mounted on a pantilt, and its tilt axis was moved from $-30°$ to $30°$, taking a scan every $1°$. One example is shown in figure 10(a). Points in the area imaged by the camera are painted with the gray color of their corresponding pixel, while points not imaged are on an uniform gray. Three such point clouds are shown registered in figure 10(b), with the other point clouds highlighted in green and blue.

C. *Final adjustment with ICP*

After the process shown in sections III-A and III-B is completed, ICP or other methods can be used to obtain a final adjustment. This was unnecessary on the outdoor sequence of figure 9. But in indoor environments, often metalic structures or eletric equipment interferes on the magnetic field, and thus the IMU compass output has larger errors. In such conditions our method can only be used as a first approximation for other



(a) A point cloud taken by a LRF

(b) Two registered point clouds.

Fig. 10. Registration of three point clouds taken by the LRF.

techniques. Averaging the processing times for 10 different pairs of point clouds, after the data is acquired, it takes 1.5 to generate interest points on both images, plus 1.5 to run the process described here, against 7 seconds to execute ICP.

IV. Conclusion

From a large number of small point clouds, a smaller number of larger point clouds were generated, by registering sequences of point clouds around a reference frame. ICP or other point cloud matching algorithms can use of the process described here as a good initial approximation, specially in applications where odometry is not available. Also, in some situations ICP may be unreliable, such as when the point clouds have low overlap, and a better initial approximation can allow ICP to avoid local minima.

The inertial data was used to eliminate the degrees of fredom associated with rotation, grounding the point clouds into a north-east-up frame of reference, and allowing the usage of a simple translation-only movement model - that allowed a single run of a robust algorithm to detect gross outliers both on the pixel correspondences and on the stereo calculations.

It is expected that the larger point clouds will be easier to register among themselves than if one had to deal directly with one point cloud per frame. This is left as future work.

V. Acknowledgements

This publication was supported by the grant BD/19209/2004 from the Portuguese Foundation for Science and Technology and by EC project BACS (FP6-IST-027140).

References

[1] Allied Vision Tech., 2007. http://www.alliedvisiontec.com/.

[2] Herbert Bay, Tinne Tuytelaars, and Luc van Gool. SURF: Speeded Up Robust Features. In *the Ninth European Conference on Computer Vision*, Graz, Austria, May 2006.

[3] J. Bouguet. Camera Calibration Toolbox for Matlab. http://www.vision.caltech.edu/bouguetj/calib_doc/index.html, 2006.

[4] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Comm. of the ACM*, 24:381–395, 1981.

[5] Heiko Hirschmuller, Peter R. Innocent, and Jon Garibaldi. Fast, unconstrained camera motion estimation from stereo without tracking and robust statistics. In *7th International Conference on Control, Automation, Robotics and Vision*, Singapore, 2 - 5 December 2002.

[6] A. Kleiner and B. Steder et al. RescueRobots Freiburg, Team Description Paper. In *Rescue Robot League*, Osaka, Japan, 2005.

[7] K. Konolige. Small vision systems: Hardware and implementation. In *8th Int. Symp. on Robotics Research*, Hayama, Japan, October 1997.

[8] J. Lobo and J. Dias. Vision and inertial sensor cooperation using gravity as a vertical reference. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 25(12):1597–1608, December 2003.

[9] J. Lobo and J. Dias. Relative pose calibration between visual and inertial sensors. In *ICRA Workshop on Integration of Vision and Inertial Sensors - 2nd InerVis*, Barcelona, Spain, April 18 2005.

[10] J. Lobo and J. Dias. Relative pose calibration between visual and inertial sensors. *Int. J. of Robotics Research*, 2007. (in press).

[11] J. Lobo, J. F. Ferreira, and J. Dias. Bioinspired visuo-vestibular artificial perception system for independent motion segmentation. In *ICVW06 (2nd Int. Cognitive Vision Workshop)*, Graz, Austria, May 2006.

[12] Jorge Lobo. InerVis Toolbox for Matlab. http://www.deec.uc.pt/∼jlobo/InerVis_WebIndex/, 2006.

[13] Jorge Lobo and Jorge Dias. Inertial sensed ego-motion for 3d vision. *Journal of Robotic Systems*, 21(1):3–12, January 2004.

[14] L. Matthies and S.A. Shafer. Error modeling in stereo navigation. *IEEE J. of Robotics and Automation*, RA-3(3), Jun 1987.

[15] L. Mirisola, J. Lobo, and J. Dias. Stereo vision 3D map registration for airships using vision-inertial sensing. In *12th IASTED Int. Conf. on Robotics and Applications (RA2006)*, Honolulu, HI, USA, August 2006.

[16] K. Ohno and S. Tadokoro. Dense 3D map building based on LRF data and color image fusion. In *IEEE Int. Conf. on Intelligent Robots and Systems (IROS 2005)*, pages 2792– 2797, Aug 2005.

[17] Davide Scaramuzza, A. Harati, and R. Siegwart. Extrinsic self calibration of a camera and a 3d laser range finder from natural scenes. In *IEEE Int. Conf.on Intelligent Robots and Systems (IROS)*, San Diego, CA, USA, Oct 2007. (submitted for publication).

[18] R. Triebel, P. Pfaff, and W. Burgard. Multi-level surface maps for outdoor terrain mapping and loop closing. In *Int. Conf. on Intelligent Robots and Systems (IROS)*, Beijing, China, October 2006.

[19] Videre Design., 2007. www.videredesign.com.

[20] XSens Tech., 2007. www.xsens.com.