Stereo Vision 3D Map Registration for Airships using Vision-Inertial Sensing

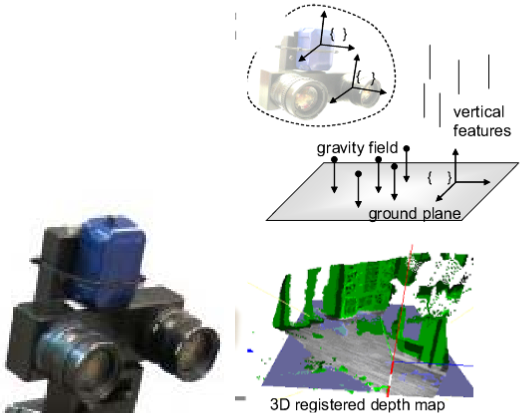
***Abstract*— A depth map registration method is proposed in this article, and experimental results are presented for long three-dimensional map sequences obtained from a moving observer.**

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 structure feature orientations. In this work we explore the fusion of stereo techniques with data from the inertial and magnetic sensors, enabling registration of 3D maps aquired by a moving observer.

The article reviews the camera-inertial calibration used, other works on registering stereo point clouds from aerial images, as well as related problems as robust image matching. The map registration approach is presented and validated with experimental results on ground outdoor environments.

I. Introduction

Inertial sensors attached to a camera can provide valuable data about camera pose and movement. Micromachining enables the development of low-cost single-chip inertial sensors that can be easily incorporated alongside the camera's imaging sensor, thus providing an artificial vestibular system.



(a) Stereo vision System (b) Frames of References.

Fig. 1. The vision-inertial system.

Figure 1 shows a stereo-camera pair with an inertial measurement unit (IMU) which is used with the aerial robotic platform shown in figure 2.

Inertial sensors provide valuable data to deal with the camera motion [-]. Visual and inertial sensing are two sensory modalities that can be explored to give robust solutions on image segmentation and recovery of 3D structure from images [-].

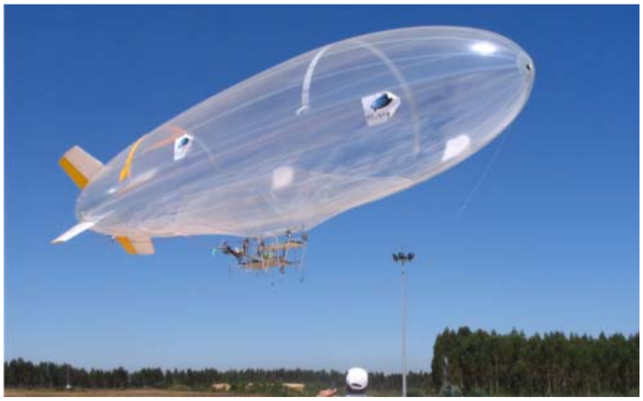


Fig. 2. The aerial vehicle instruments include a vision system and a low-cost inertial measurement unit.

Perception systems for robotic applications have since recently taken advantage of low-cost inertial systems to complement vision systems. Biological vision systems also utilise inertial cues, such as provided by the vestibular system, often in the early stages of image processing.

An inertial sensor coupled with a camera provides direct measures of the camera orientation, and with magnetometers and accelerometers (that measure gravity), these orientation measures are grounded on the world north-east-up frame of reference. These measures can be combined with measures taken from the vision system, to simplify tasks such as the reconstruction of the observed world, or to improve their accuracy.

Depending on the level of accuracy desired, and on the quality of inertial measurements, the inertial measurements can eliminate some degrees of freedom from vision-based estimation tasks, or at least provide a good initial approximation, therefore allowing faster processing or the use of simpler movement models.

Stereo vision systems can use correlation based methods to obtain depth maps. With the current technology, real time systems are commercially available [1]. When the vision system is moving the maps have to be fused into single world map. Before fusing the depth maps, they must be registered to a common referential. This can be done using data fitting alone, or aided by known parameters or restrictions on the way the measurements were made.

In our work, correlation based stereo depth maps are obtained by the moving vision system, and rotated to a common levelled reference provided by the rotation update from inertial sensed gravity and magnetic sensed bearing.

But there remains a 3D translation in the successive depth maps due to the motion, for which the inertial sensors only provide a rough estimate. By tracking some image targets over successive frames, the system translation between frames can be estimated by subtracting their 3D position. Fully registered depth maps can therefore be obtained from the moving system.

In [2], a stereovision only aproach is presented to build a 3D map of the environment from stereo images taken by a remotely controlled airship, and at the same time localizing the vehicle (what it is known by SLAM). The system keeps a Kalman Filter where its state vector contains the camera pose and the position of automatically detected landmarks on the ground. Their vision system computes a 3D point cloud for each stereo image pair, and matches interest points between successive images. Some of the interest points are selected to be used as landmarks - their positions are included on the Kalman Filter state and their detection is used as a measurement to Kalman Filter update.

The strengths of their work include: the interesting point detection and matching algorithm[3], based on finding affine transformations to match a small group of interest points, and then use the newly found transformation to focus the matching of other groups, until a transformation is found that can match enough interesting points; the determination of the stereo vision error, that affects the position of individual 3D points and of the landmarks; and the treatment of uncertainty - for every measurement of landmarks, motion estimation, or Kalman filter update, there is at least a reasonable approximation for the uncertainty, avoiding an empirical “filter tuning” stage.

They achieved centimeter level accuracy with a baseline/depth ratio of aproximatelly $1/15$ (depth corresponds basically to altitude). As they point out themselves, data from other sensors might be integrated on their framework, but this was not their aim at the time.

In the work reported in this article, the image registration starts by using the inertial measurements of the camera orientation to rotate the 3D point clouds obtained from stereo to a common orientation, aligned with the north-west-up frame of reference. There remains a 3D translation between successive point clouds, due to the camera motion, but the inertial system can only provide a rough estimate of it.

Then, point correspondences on the image space are utilized to find the translation that registers the point clouds: taking into account only the pixel correspondences that refer to corresponding 3D points, each pair of corresponding 3D points yields a direct measure of the translation between the point clouds. There are outliers, that are excluded with a robust algorithm. A single translation vector is calculated by averaging the inliers and the point clouds are translated into an unified frame of reference.

We leverage on previous work that calibrate the rigid body rotation between a camera and a inertial system that are rigidly coupled, to register a set of 3D point clouds taken from a moving observer with an calibrated stereo head.

In the future, the moving observer will be the aerial vehicle. The image sequences shown here were captured within real enviroments, including scenes that mimic, in small scale, forest and buildings.

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 point cloud a limited sequence of neighbouring point clouds around one taken as reference.

This unified aggregated 3D point cloud potentially has a lot of redundant points. To reduce memory usage, a hash table allows us to search for existing points that are too close to a new point, and to reject the new point if it is redundant. The redudant points are also used to eliminate gross stereo errors - filtering out points that do not appear in more than a minimum number of point clouds.

From a large number of smaller point clouds, we can therefore construct a smaller number of larger point clouds, with many redundant or grossly wrong points eliminated, that should be easier to register between themselves, what is left to future work.

The next section reviews the camera-inertial calibration to be used, other works on registering stereo point clouds from aerial images, as well as related problems as robust image matching. Section II describes our present approach, followed by experimental results on section V and finally the conclusions on section VI.

II. Registering Stereo Depth Maps

A moving stereo observer of a background static scene with some moving objects can compute at each instant a correlation-based dense depth map. The maps will change in time due to both the moving objects and the observer ego-motion. To perform independent motion segmentation, a first step in processing the incoming data is to register the maps to a common fixed frame of reference $\{\mathcal{W}\}$, as shown in figure 3.

The stereo cameras provide intensity images $I_l(u,v)|_i$ and $I_r(u,v)|_i$, where u and v are pixel coordinates, and i the frame time index. Having the stereo rig calibrated, depth maps for each frame can be computed. A set of 3D points ${}^{\mathcal{C}}\mathbb{P}|_i$ is therefore obtained at each frame, given in the camera frame of reference $\{\mathcal{C}\}|_i$. Each 3D point has a corresponding intensity gray level c given by the pixel in the reference camera, i.e $c = I_l(u,v)|_i$. Each point in the set retains both 3D position and gray level

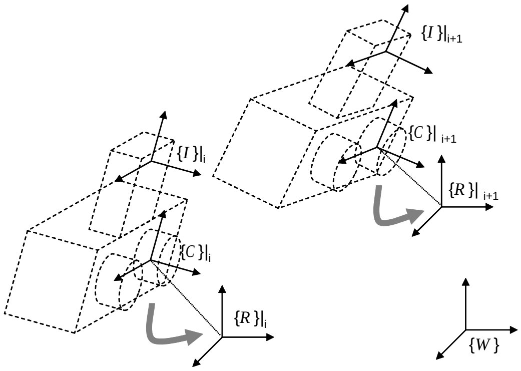


Fig. 3. Moving observer and world fixed frames of reference.

$$P(x, y, z, c) \in {}^{\mathcal{C}}\mathbb{P}|_i \ . \tag{1}$$

A. Rotate to Local Vertical and Magnetic North

The inertial and magnetic sensors, rigidly fixed to the stereo camera rig, provide a stable camera rotation update ${}^{\mathcal{R}}\boldsymbol{R}_{\mathcal{C}}$ relative to the local gravity vertical and magnetic north camera frame of reference $\{\mathcal{R}\}|_i$.

Calibration of the rigid body rotation between $\{\mathcal{I}\}|_i$ and $\{\mathcal{C}\}|_i$ can be performed by having both sensors observing gravity, as vertical vanishing points and sensed acceleration, as described in [4].

The rotated camera frame of reference $\{\mathcal{R}\}|_i$ is time-dependent only due to the camera system translation, since rotation has been compensated for.

B. Translation from Image Tracked Target

The translation component can be obtained by tracking a fixed target in the scene. The tracked image feature must have the corresponding 3D points P_t in each depth map, so that the translation can be estimated from

$$\Delta\vec{t} = \boldsymbol{P}_t|_{i+1} - \boldsymbol{P}_t|_i \tag{2}$$

with $\boldsymbol{P}_t|_{i+1} \in {}^{\mathcal{R}}\mathbb{P}|_{i+1}$ and $\boldsymbol{P}_t|_i \in {}^{\mathcal{R}}\mathbb{P}|_i$.

The fixed target can be an artificial one, or a set of sparse natural 3D features can be tracked to improve robustness. Distinctive image features can be automatically obtained with the Scale Invariant Feature Transform (SIFT [5]). Assuming that the majority of distinctive features are from the static background, random sample consensus (RANSAC [6]) can be used to reject outliers that occur from tracking features of the moving objects or due to errors on the process.

III. Results

The hardware system used to acquire data from a moving observer is shown in fig. 1. The stereo vision is provided by the Videre MEGA-D Digital Stereo Head [7], and the pose from the inertial and magnetic sensor package MT9-B from Xsens [8].

To compute range from stereo images we are using the SRI Stereo Engine with the Small Vision System (SVS) Software [1].

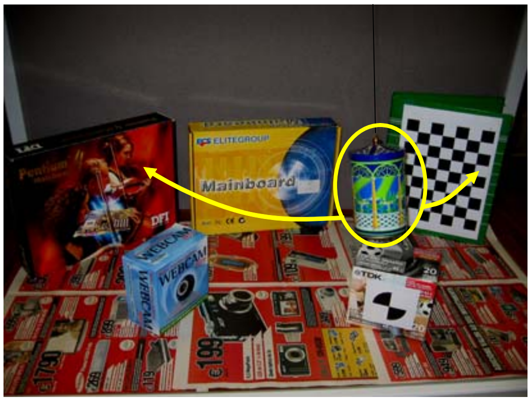


Fig. 4. Experimental setup of 3D scene with static background and swinging pendulum.

A scene was set up with a swinging cylindrical can to provide motion independent from the observer movement (fig. 4). The moving observer surveyed the scene performing map registration and subsequent independent motion segmentation as presented in the following sections.

A. Moving Depth Map Registration

As described above, the rotation update provided by the inertial and magnetic sensor package is applied to the successive depth maps. As shown in figure 5, the depth maps are correctly rotated, but shifted due to the observer translation.

The translation was estimated by tracking an image feature, and observing the translation between the corresponding 3D points in the depth maps. Figure 5 shows data for frames 1 and 20 of a take of 200 frames with a moving observer of a static scene with a moving pendulum, for which the registration performed well.

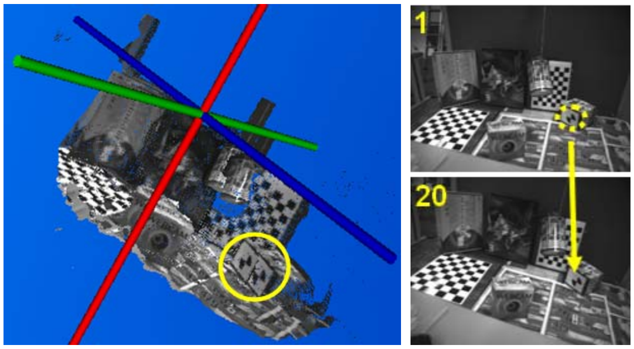


Fig. 5. Overlaid rotated 3D depth maps from frames 1 and 20 (on the right) showing a clear mismatch, and circled image feature tracked to estimate translation.

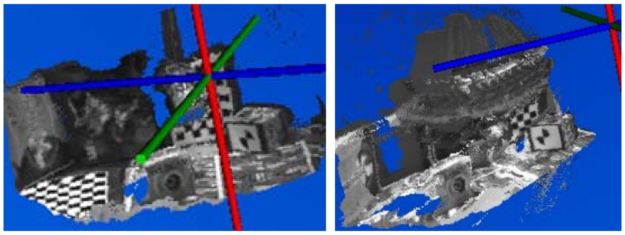


Fig. 6. Depth maps rotated and translated to common world fixed frame of reference, for frames 1 and 20 on the left, and for full set of frames with moving pendulum on the right.

The registered depth map can be seen in figure 6. The fused map from frames 1 and 20 is shown on the left. On the right the fused map corresponding to the full set of frames is shown with the moving pendulum leaving its trace.

IV. Registering 3D Point Clouds

A. Obtaining rotation and translation measurements between two point clouds

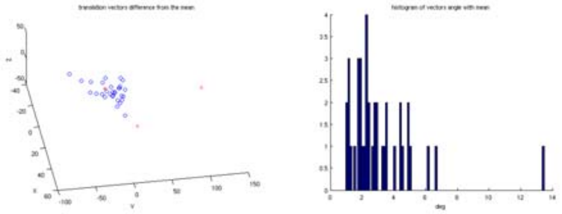
We record a sequence of stereo image pairs and associated inertial data taken from a moving observer, and generate a 3D point cloud for each stereo image pair i, that is generated into the camera frame of reference $\{\mathcal{C}\}|_i$. To register these point clouds, it is necessary to compensate for the rotation and translation of these cameras in relation to a reference coordinate system.

We take advantage of recent camera-inertial system calibration, to obtain absolute orientation measurements for the camera, that compensate for the rotation of the camera, and obtain rotated 3D point clouds that differ only in translation.

The raw image frames are the other source of information to find the displacement between these point clounds. Given two images i and j (taken in different instants), with an overlap region, it is possible to find corresponding points between these two images. As the stereo processing maps image pixels to a 3D points, the corresponding image points should map to corresponding 3D points in the $\{\mathcal{R}\}|_i$ and $\{\mathcal{R}\}|_j$ frames of reference. Stereo algorithms may skip some image points, not mapping them to a 3D point, (e.g., due to lack of texture). Therefore we consider only the image point correspondences whose pixel coordinates are mapped to a 3D point in both images.

As the two rotated point clouds have already the same orientation, the diference in the coordinates of the corresponding 3D points is a measure of the translation vector that brings the reference frame of one rotated point cloud $\{\mathcal{R}\}|_i$ to the other $\{\mathcal{R}\}|_j$. Additionally, given that the origins of the $\{\mathcal{R}\}$ reference frames are the camera centers, the translation vector also corresponds directly to the coordinates of one camera center into the frame of reference of the other camera.

The SIFT feature detector [9] was used to detect point correspondences, and the RANSAC[10] procedure was applied to reject outliers. RANSAC is applied to the 3D translation vectors (and not on the pixel coordinates of the correspondences), looking for a inlier set where all vectors are within a maximum distance of their average (a few centimeters in our case). Both mismatched SIFTs 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 vector and the mean vector - i.e., if all vectors were the same, all circles would appear on the origin. The 'x's are outliers, that were detected by RANSAC and excluded from the calculations. The '+' signal is the mean of the inliers. Figure 7(b) is a histogram of the angle between the translation vectors and the mean vector, showing that most of them point approximatelly to the same direction, except a few outliers corresponding to the crosses on the left figure.



(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. One example of the usage of the RANSAC procedure.

As the model utilised on the RANSAC calculations is very simple, involving averaging and calculating differences between euclidean 3D vectors, the RANSAC procedure runs very fast, on the order of a few tenths of seconds per frame in MATLAB.

As there is not an absolute frame of reference as a GPS, we can choose arbitrarely one reference frame $\{\mathcal{R}\}|_0$ as the global frame where the other point clouds will be registered to.

B. Filtering out redundant points

We are registering point clouds that may have a large overlap, and therefore may have a large number of 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 would be too slow to check linearly all the stored points to test if a new point is redundant.

Additionally, it is necessary to filter the point clouds, as there are often wrongly positioned points due to errors on the stereo processing. We aim to eliminate isolated, "floating" points, and 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 voxel approach is that the number of voxels increases with the covered space, and many voxels are empty, outside and inside the 3D surface visible on the scene. It is desirable to avoid this waste of memory to be able to cover a larger space.

We have chosen the well-known approach of keeping only a 3D point cloud and a hash table, that indexes all points by their coordinates. When a new point is going to be inserted, the hash table is used to retrieve a list of potentially close points, and a close point is searched for.

If there are repeated passages over the same scene, we can filter out doubtfull points by deleting points that were not seen in a sufficient number of frames. To keep track of this, every point in the cloud 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.

V. Experimental Results

A. Experimental Platform

The experimental platform is a Videre MEGA-D digital stereo head [11], rigidily coupled with the inertial sensor package MTB9-B from Xsens[12].

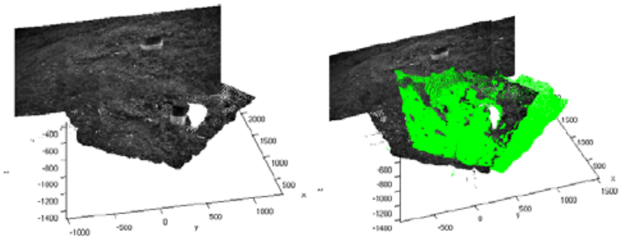
Xsens inertial measurements are taken immediatelly after each frame aqcuisition. To compute range from stereo images we are using the SRI Stereo Engine with the Small Vision System (SVS) software [13]. This platform was moved by hand, generating data sets consisting of stereo image pairs, point clouds, and inertial measurements.

B. Experiments

Given a sequence of image frames, inertial measurements and 3D point clouds, an image frame is chosen as the reference frame, and the frames immediatelly before and after it in time are matched with it using the usual SIFT matching process. Each pair of matching SIFT features whose pixel location corresponds to a 3D point on both images yields a translation vector, by subtracting the coordinates of the respective 3D points. Therefore a set of translation vectors is generated for a given image pair, measuring of the displacement between the two camera frames of reference $\{\mathcal{C}\}|_i$ and $\{\mathcal{C}\}|_j$, and the RANSAC procedure search for a consensus translation vector on this set.

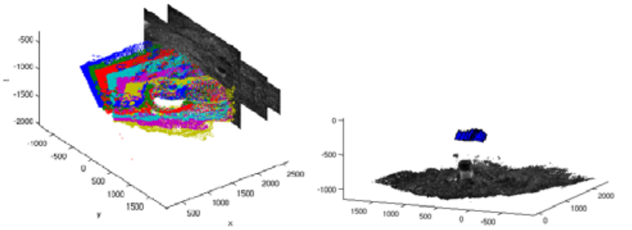
As many backward and forward frames are matched as it is possible, by keeping trying to match frames in the sequence untill the number of outliers falls below a certain threshold - probably because the camera has moved and the overlap region is too small.

The garden sequence was taken on an outdoor grass field, under indirect sunlight. The sift algorithm can extract a large number of features from the grass surface, that is roughly planar, although irregular. Figure 8(a) shows one point cloud from this data set, with its corresponding left camera image displayed on the back. Figure 8(b) shows two registered point clouds, one in plain green, other in the original gray level color. Figure 9 shows the resulting point cloud after registering together a sequence of 26 successive frames.



(a) (b)

Fig. 8. From the garden dataset: (a) one point cloud; (b) two registered point clouds.



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

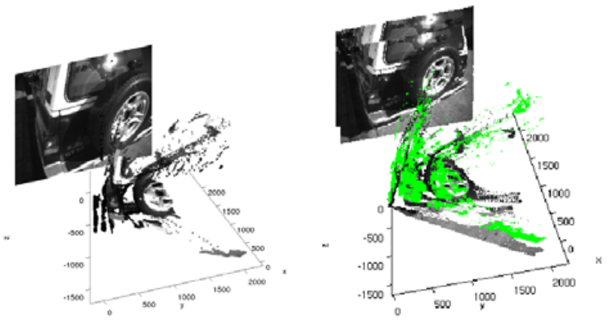
(b) the final, filtered point cloud. The camera trajectory is shown in blue.

Fig. 9. The result of registering point clouds for 26 successive frames.

The jeep sequence shows a jeep parked on the street under sunlight, and the sidewalk. Analogously, Figures 10(a) and 12(a) show one point cloud from this data set, with its corresponding left camera image displayed on the back. Figures 10(b) and 12(b) show two registered point clouds, in the original gray level color, with their respective images displayed behind them. Figures 11 and 13 show, in the left, a set of registered point clouds, and in the right, the resulting point cloud after registering together a sequence of 27 (in both figures) successive point clouds, and filtering out points with less than four votes. In the left figure, only one every four point clouds is shown, to ease visualisation.

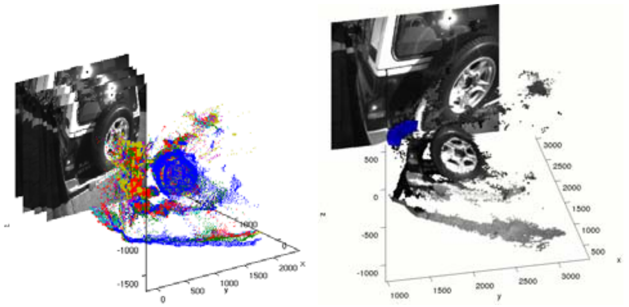
The pyramids on figures 9, 11 and 13 represent the camera trajectory and orientation: there is one pyramid for every four camera poses, and camera points towards the base of the pyramid. The images on the graph sides are an approximate visual reference, being stretched up to the extreme coordinates of their point clouds.

The RANSAC threshold for membership in the inlier set was $5\,cm$ for both datasets. The minimum acceptable number of inliers was 20.



(a) (b)

Fig. 10. From the jeep dataset: (a) one point cloud; (b) two registered point clouds.



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

(b) The resulting, filtered point cloud. The camera trajectory is shown in blue.

Fig. 11. The result of registering point clouds for 27 successive frames.

For each frame chosen as a reference frame, we could register between 20 and 50 frames, (approximatelly between 1.5 and 3 seconds at 15fps).

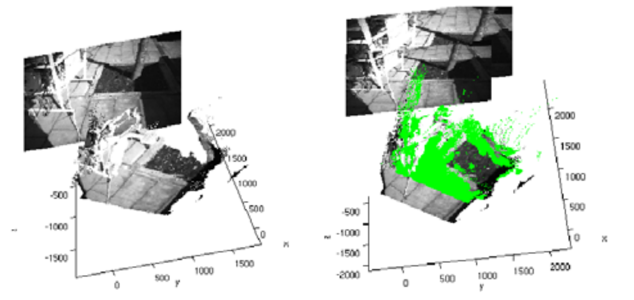
VI. Conclusion

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.

If greater accuracy were desired, more sofisticated point cloud matching algorithms could make use of the process described here as an initial approximation.

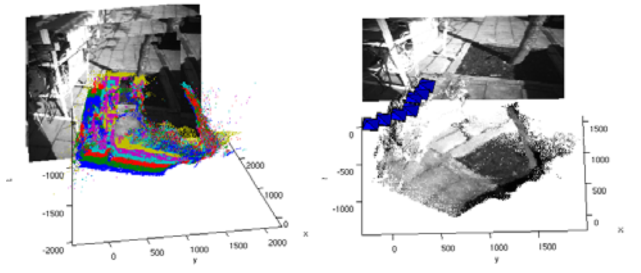
The inertial data was used to eliminate the degrees of fredom associated with rotation, grounding the point clouds into a north-east-up frame of reference, and allowing the usage of a simple translation-only movement model - that allowed a single run of a robust algorithm to detect gross outliers both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

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



(a) (b)

Fig. 12. The sidewalk from the jeep dataset: (a) one point cloud; (b) two registered point clouds.



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

(b) The resulting, filtered point cloud. The camera trajectory is shown in blue.

Fig. 13. The result of registering point clouds for 27 successive frames.

References

[1] Kurt Konolige. Small vision systems: Hardware and implementation. In *Eighth International Symposium on Robotics Research*, Hayama, Japan, October 1997.

[2] E. Hygounenc, I-K. Jung, P. Soueres, and S. Lacroix. The Autonomous Blimp Project at LAAS/CNRS: Achievements in Flight Control and Terrain Mapping. *International Journal of Robotics Research*, 23(4/5):473–512, April/May 2004.

[3] I-K. Jung and S. Lacriox. A robust interest point matching algorithm. In *8th International Conference on Computer Vision*, Vancouver, Canada, July 2001.

[4] Jorge Lobo and Jorge Dias. Relative pose calibration between visual and inertial sensors. In *ICRA 2005 Workshop on Integration of Vision and Inertial Sensors (InerVis2005)*, Barcelona, Spain, April 2005.

[5] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[6] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981.

[7] Videre Design. http://www.videredesign.com/.

[8] Xsens Technologies. http://www.xsens.com/.

[9] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[10] 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.

[11] Videre design., 2005. www.videredesign.com.

[12] Xsens technologies., 2005. www.xsens.com.

[13] Kurt Konolige. Small vision systems: Hardware and implementation. In *Eight International Symposium on Robotics Research*, Hayama, Japan, October 1997.