CO-REGISTRATION OF KINECT POINT CLOUDS BASED ON IMAGE AND OBJECT SPACE OBSERVATIONS

Ali Khosravani[1], Maurice Lingenfelder[2], Konrad Wenzel[1] and Dieter Fritsch[1]
Institute for Photogrammetry, University of Stuttgart, Germany
Email: [1]:[firstname.lastname]@ifp.uni-stuttgart.de, [2]:maurice@lingenfelders.de

KEY WORDS: Multi-Sensors, Low-cost, Acquisition, Registration, Kinect, KinectFusion, Structure from Motion

ABSTRACT:

In this paper, two approaches for the alignment of point clouds from the RGB-Depth sensor *Microsoft Kinect*, based on image and object space observations are described and evaluated. The first approach is based on the RGB images and estimates a sensor pose using image features, while the second one uses only the geometrical information provided by the range data. For the image-based method, *Structure from Motion* methods are used, which incrementally estimate relative orientations between images based on feature points and merges the solution in a bundle adjustment. For the object space-based method, the depth image is used within the software *KinectFusion*, which estimates surfaces by volumetric range image integration and determines the sensor pose by the *Iterative Closest Point* algorithm (ICP). Within this paper both methods are evaluated and compared regarding their performance in the estimation of the sensor pose. Also, an application of these registration methods is presented, where the sensor pose is used in combination with two additional cameras in order to retrieve high density point clouds by means of dense image matching.

1. INTRODUCTION

During the last two years, the low-cost sensor system Microsoft Kinect has been widely considered by many researchers for a variety of indoor applications, in particular for mapping and navigation. The system consists of an infrared (IR) laser projector, an RGB and a monochrome IR CMOS sensor. The depth measurement is realized by projecting a known pseudo-random speckle pattern onto the object surface using the IR laser. The IR camera acquires the pattern in 30Hz at VGA resolution (640 by 480 pixels). By analysis of the IR speckle pattern, automatic stereo measurement is realized, in order to compute a range image (with 11 bit depth) from spatial intersection using the relative orientation between projector and IR camera. The system angular field of view is 57° horizontally and 43° vertically. The Kinect system has a practical ranging limit of 1-3.5m, although the sensor can maintain tracking through an extended range of about 0.7-6m (Wikipedia: Kinect, 2012).

The relative orientation between the RGB and the depth image obtained from the stereo calibration of the system can be used to provide a co-registration of multiple 3D point clouds. For this purpose, sparse image features are extracted from the RGB images and matched. Subsequently, the camera orientations can be estimated and refined within a bundle adjustment. The scale information can be derived from the distance information for the feature points stored in the depth image, since the relative orientation between the RGB and the metric depth image is known.

In addition, the point clouds or range images can be co-registered directly using the geometrical information. The approach we describe here is based on Iterative Closest Point (ICP) algorithm (Besl & MacKay, 1992) for the alignment of the current Kinect range image with the previous frames. An efficient implementation of this approach, so called KinectFusion, is developed by Microsoft Research Group, which is able to reconstruct the scene and to continuously track the Kinect sensor in real-time, based on GPU implementation of a coarse-to-fine ICP algorithm (Izadi et al., 2011). Here, we use the open source implementation of KinectFusion provided by the Point Cloud Library (PCL).

Within the paper, the image-based and the object space-based method for estimating the sensor pose will be described and evaluated. Also, an application will be demonstrated, in which highly accurate dense point clouds are computed using two high resolution industrial cameras in combination with the Kinect system. The dense point clouds are derived using a dense image matching implementation by the Institute for Photogrammetry. The orientation registration of the point cloud for each shot can be provided by KinectFusion. Since both images are exposed at the same time, the speckle pattern projected by the Kinect can be used as artificial texture. Thus, no object texture is required for data acquisition with this sensor.

2. POINT CLOUD ALIGNMENT USING RGB IMAGES

The Kinect system is a RGB-D sensor that captures RGB images along with per-pixel depth information. Similar to the work of Henry et al. (2010), the alignment of the Kinect point clouds is efficiently realized in this study firstly by automatic relative orientation of consecutively captured RGB images.

In order to make use of the information provided by the RGB images for this purpose, it is necessary to find the relative orientation between the RGB and disparity image. In other words, by stereo calibration of RGB and IR cameras, the correspondence information between a pixel in the depth image and in the RGB images can be determined, while simultaneously compensating the lens distortion effects.

In this work, the RGB images were relatively oriented using the solution presented by Abdel-Wahab et al. (2011). They describe the solution by the following main steps:

(a) Fast image indexing to avoid time consuming matching of all possible image pairs.

(b) Generating tie points by means of automatic feature extraction and matching (e.g. SIFT feature points (Lowe, 2004)).

(c) Detecting reliable patches of images having mutual compatibility, and optimizing the geometry graph for each patch to ensure optimization of the final spanning tree.

(d) Merging all patches and finally adjusting the model.

The core of this pipeline is the Structure from Motion (SfM) approach, which is used for the derivation of exterior orientation parameters, which serve as initial values for the final bundle adjustment. The scale factor can be estimated, having the 3D coordinates of some of the SIFT features in the RGB images, using the abovementioned correspondence information between a pixel in the depth image and in the RGB images.

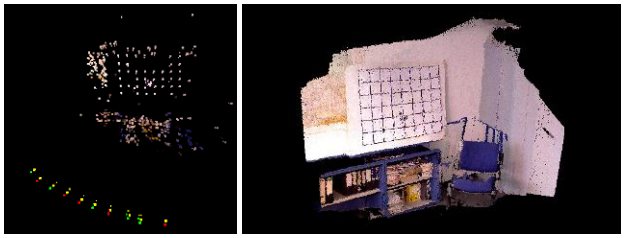

Figure 1 - Camera poses and triangulated feature points (left), co-registered point clouds (right)

Figure 1 shows an exemplary output of this approach for the alignment of point clouds collected from 10 viewpoints. The adjusted values for the camera poses are directly used for the alignment of the point clouds.

3. POINT CLOUD ALIGNMENT IN OBJECT SPACE

The previous approach for the alignment of the point clouds relies on the existence of sufficient visual features in the scene, since reliable orientation parameters can only be determined if enough feature points are available. Sparsely textured areas will either lead to low accuracy or even to a failure in the orientation of images.

In contrast to the first approach, the following approach presented here relies on the geometrical information provided by the range images. To align the point clouds in this approach, the camera poses are estimated for each new frame using the ICP algorithm.

KinectFusion

In 2011, Microsoft Research Group developed a new application for Kinect, so called KinectFusion. As described by Izadi et al. (2011), KinectFusion allows a user holding and moving a standard Kinect system, to reconstruct an indoor scene in 3D rapidly, while continuously tracking the 3D pose of the sensor. The 3D pose of the sensor is estimated in real-time by registering the new depth data to the previously extracted surface, using a GPU implementation of a coarse-to-fine ICP algorithm (Newcomb et al. 2011).

Moreover, the software integrates the range images using a volumetric method, in order to retrieve low noise surfaces using the available redundancy from the high number of depth images. This step follows the approach presented by Curless and Levoy (1996). Therefore, a truncated signed distance field is used to fuse the data. Subsequently, the meshed surface is extracted from this field and rendered in real-time. Thus, visual feedback is provided during the scanning process. Figure 2 compares a single depth frame from the Kinect with the output of KinectFusion.

Izadi et al. (2011) describe the main system pipeline by the following four main stages. These steps are executed in parallel on the GPU for the real-time capability of the system.

a) *Depth Map Conversion*: The range image is converted to a 3D point cloud and the normal vector is estimated for each point.

b) *Camera Tracking*: In this phase, the current point cloud is aligned to the previous data, using a GPU implementation of the ICP algorithm. In fact, the scene motion computed by the ICP algorithm is equivalent to the camera motion.

c) *Volumetric Integration*: The registered range image is integrated into a volumetric voxel space, following the approach of Curless and Levoy (1996). Thus, all measurements can be subsequently considered to extract an optimal surface from the redundant observation.

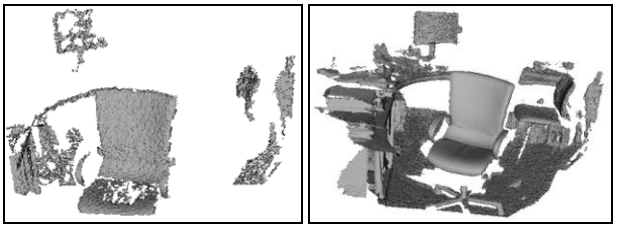

Figure 2 - Single depth frame (meshed point cloud, left) versus the output of KinectFusion (right). The holes in the point clouds are filled and the noise is removed

d) *Raycasting*: Finally, the views of the implicit surface are generated using a GPU-based raycaster, for the rendering and tracking purpose.

In this study we used an open source implementation of KinectFusion provided within the Point Cloud Library (PCL).

4. EVALUATION: ACCURACY OF POSE ESTIMATION USING THE DESCRIBED METHODS

In order to evaluate the performance of the image-based and the object space-based method in indoor applications, the accuracy of the pose estimation was determined using accurate reference data. This reference pose was acquired using a calibration pattern in combination with a bundle adjustment within the software Australis.

To evaluate the sensitivity of the two methods for registration regarding the existence of visual and 3D features, two scenarios were acquired. The first scenario has more texture and less 3D features, whereas the second one has less texture and more 3D features (figure 3).

The image-based registration method based on SfM does not provide scale information. In order to avoid the introduction of an error resulting from the estimation of the scale factor using the Kinect depth image in this analysis, the camera positions were fitted to the reference positions using a 7 parameters similarity transformation.

Table 1 indicates that in the first scenario, the SfM method delivers relatively better results for the pose estimation, whereas in the second scenario, the accuracy delivered by KinectFusion is considerably higher. Having considered the table, we can deduce that firstly, the amount and distribution of visual features in the scene plays an important role for the SfM method. Scenario 1 depicts an example of an indoor scene with plentiful visual features; however, the accuracy of the estimated camera pose is not considerably high. The situation gets even more critical in the second scenario, which is still a typical indoor scene. Consequently, this method can easily fail in scenes with less texture like corridors and most of the public buildings' interiors.

Secondly, we can deduce from the table that the amount of 3D features in the scene is very important for the KinectFusion software. The table shows that in the first scenario, compared to the second one, the camera pose cannot be estimated with a high accuracy, as that the ICP algorithm cannot accurately fix the camera pose 6 DOF in this scene. Adding some 3D features to the scene (second scenario), results in significantly higher accuracy in the camera pose estimation by this method.

This shows that the choice between the two methods highly depends on the existence of enough visual and 3D features. Of course, to achieve a more reliable and more accurate solution, one should take the advantages of both methods.

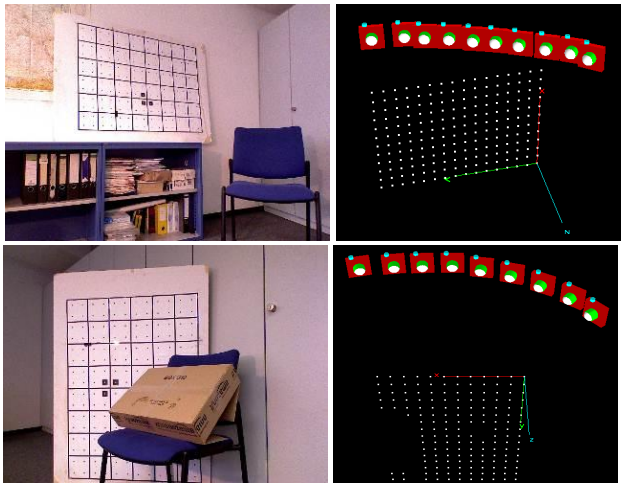

Figure 3 - Scenario 1: more texture, less 3D features (top)
Scenario 2: less texture, more 3D features (bottom)

RMS [mm]	SfM		KinectFusion	
	Scenario 1	Scenario 2	Scenario 1	Scenario 2
ΔX	4.4	9.8	13.2	1.8
ΔY	8.7	6.1	21.9	2.6
ΔZ	4.3	10.0	13.7	2.2
Dist.	10.6	15.3	29.0	3.9

Table 1 - Accuracy of camera pose estimation

5. APPLICATION: GENERATING HIGH ACCURACY DENSE POINT CLOUDS

The KinectFusion is able to acquire surface information in real-time due to the GPU implementation of the main pipeline. The quality and density of the point cloud generated by KinectFusion is dramatically higher than other SLAM systems, as they just focus on dense tracking, and use sparse maps for the localization purpose (Izadi et al., 2011). However, this quality is not as high as the state-of-the-art offline registration and surface reconstruction algorithms (e.g. dense image matching) using high resolution cameras.
Fritsch et al. (2011) present an image-based approach, where dense image matching techniques are used with a compact and affordable rig of 5 off-the-shelf industrial cameras. This enables a one-shot solution for high accuracy and dense data acquisition. As the image matching technique might fail for the reconstruction of surfaces with low texture, an additional structured light pattern generated by the Kinect is used. Therefore, it enables data acquisition and image matching in low light conditions or for featureless objects.
Similar to that work, in this study a stereo high resolution camera rig was integrated to the Kinect system to enable image matching and dense surface reconstruction. Although in this work, instead of the SfM approach for the camera rig pose estimation, the KinectFusion implementation of the PCL was employed for the determination of the camera rig pose. Thus, dense data acquisition is possible for environments without texture.

5.1 Multi-Sensor System

The sensor system consists of two high resolution monochrome CCD cameras (5 Megapixels) for the dense image matching and a Kinect for providing artificial texture by the speckle pattern and tracking the sensor pose by the KinectFusion.

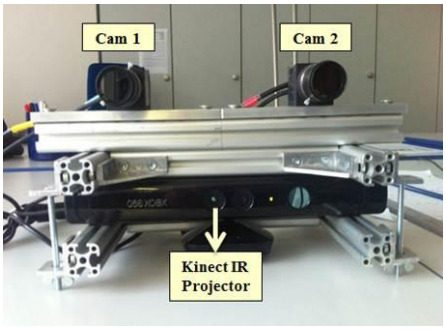

Figure 4 - Configuration of the sensor system

The two high resolution cameras are firmly mounted on top of the Kinect to maintain a stable configuration. To make the Kinect IR pattern visible to the two high resolution cameras, a 670nm daylight blocking filter was installed in front of each camera lens. The base-line of the matching cameras is set to 16cm, which leads to the estimated precision described in table 2, assuming the precision of the dense matching algorithm being around 0.3 pixels.

Distance [cm]	Predicted Accuracy [mm]	Resolution [pix/mm²]	Footprint [cm]
80	0.5	8.4	68 x 71
100	0.8	5.4	90 x 88
120	1.2	3.7	111 x 106
150	1.8	2.4	142 x 132
200	3.2	1.3	195 x 177

Table 2 - Approximated precision of camera pose estimation

5.2 Software System Overview

The data acquisition is done by holding and moving the sensor system around the object and taking suitable amount of images by the stereo cameras. The collected dataset contains stereo image pairs, together with the corresponding exterior orientations. The exterior orientations are derived by KinectFusion, using the pose estimated for the depth camera and the relative orientation between the Kinect IR camera and the matching cameras. This relative orientation as well as the interior orientation of the cameras is determined using standard calibration methods employing a calibration pattern.
The collected data then could be directly used by the software system implemented by Wenzel et al. (2011), to derive a dense and accurate point cloud. The main software pipeline consists of the following steps:
a) Dense image matching: In this step, correspondences between each stereo image pair are determined using a modified version of *Semi Global Matching* (Hirschmüller, 2008).
b) Triangulation: The output of the last step is a disparity map describing the pixel-to-pixel correspondences between stereo image pairs. In this step, the 3D point coordinates for each of these correspondences are retrieved by reconstructing the rays, having known the exterior orientation of the images.
c) Post processing: Filtering methods are applied on the resulted point cloud to remove remaining outliers.
Figure 5 shows an exemplary point cloud determined by this software, which is a collection of 3.6 million points captured from 7 sensor positions, at the distance of about 1m. However, outliers can occur for the stereo camera since each point has only 2 measurements and thus cannot be verified. An improvement is possible by refining the registration, e.g. using

the ICP algorithm on the high resolution point clouds, and applying filters using the resulting redundancy in object space.

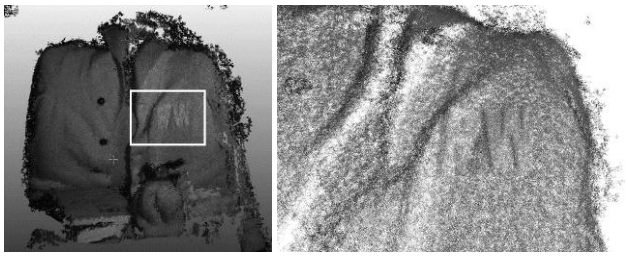

Figure 5 - High accuracy dense point cloud captured from 7 sensor positions using the described application

The acquisition with KinectFusion fails in scenes with insufficient amount of 3D features, since the ICP fails to fix the sensor pose 6 DOF. Figure 6 shows an example, in which a chessboard is captured from two different sensor positions with the distance of 10cm. As expected, a shift along the plane is visible. Such ambiguities could be resolved by integrating the image-based method of registration.

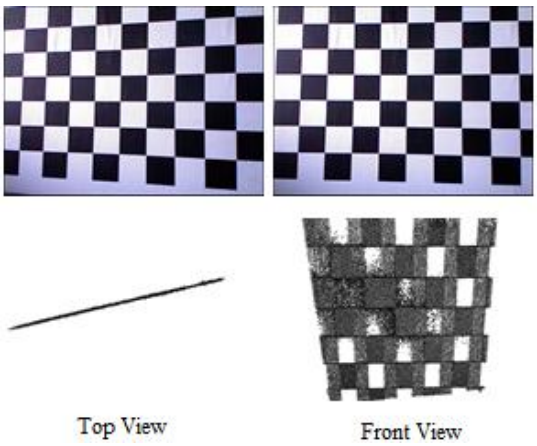

Figure 6 - The point clouds generated by this method show a misalignment in the 2D plane space

6. CONCLUSIONS AND FUTURE WORKS

Within the paper, two approaches for the alignment of Kinect point clouds, based on image and object space observations were described and compared in two different scenarios.

The results show, that the image-based registration method is particularly suitable for scenes with texture, while the object space-based method can be used on scenes without texture. Furthermore, the object space-based method requires a sufficient amount of geometric information in the scene, while the image-based method is not dependent on that. This complementary behavior leads to the conclusion, that both methods should be combined in future work, in order to provide a highly reliable method for a variety of applications. Furthermore, feature points from both methods could be used within one bundle adjustment, which provides additional quality information and enables loop-closure for the reduction of drift.

However, there are cases in which none of the described methods work properly, e.g. in corridors or along planar features with insufficient texture. In such cases, the ICP algorithm cannot align the point clouds accurately, and also the point-features based SfM approach fails to orient the corresponding RGB images. Therefore, future work is required to support the pose estimation by extraction and matching of line features in both image and object spaces.

We also presented an application in which dense point clouds were acquired using additional cameras in combination with a sensor pose determined in object space. This enables the extraction of high resolution point clouds using dense image matching without relying on surface texture, but using the speckle texture projected by the Kinect. An extension to this work may support the pose estimation task by tracking the extracted image features, similar to most of the SLAM approaches, in order to use the information from the texture, if available. Furthermore, refinements for the pose estimation are possible by using the dense point clouds within an accurate alignment. An additional volumetric integration of the point clouds could eliminate remaining outliers and reduce noise at the surface.

7. ACKNOWLEDGEMENTS

We gratefully acknowledge the contributions of Mohammed Abdel-Wahab for the orientation of the images using the described pipeline.

8. REFERENCES

Abdel-Wahab, M., Wenzel, K. and Fritsch D., 2011. Reconstruction of Orientation and Geometry from Large Unordered Image Datasets for Low Cost Applications. Low-Cost 3D (LC3D) workshop, Berlin, December 2011.

Besl, P. J. and MacKay, N., 1992. A Method for Registration of 3-D Shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence. 14(2), pp. 239-256.

Curless, B. and Levoy, M., 1996. A Volumetric Method for Building Complex Models from Range Images. ACM Trans. Graph., 1996.

Fritsch, D., Khosravani, A. M., Cefalu, A. & Wenzel, K., 2011. Multi-Sensors and Multiray Reconstruction for Digital Preservation. Photogrammetric Week '11, Ed. D. Fritsch, Wichmann, Berlin/Offenbach, pp. 305-323.

Henry, P., Krainin, M., Herbst, E., Renand, X., Fox, D., 2010. RGB-D Mapping: Using Depth Cameras for Dense 3D Modeling of Indoor Environments. RSS Workshop on Advanced Reasoning with Depth Cameras, 2010.

Hirschmüller, H., 2008. Stereo Processing by Semi-Global Matching and Mutual Information. IEEE Transactions on Pattern Analysis and Machine Intelligence, 30(2), pp. 328-341.

Izadi, S., Kim, D., Hilliges, O., Molyneaux, D., Newcombe, R., Kohli, P., Shotton, J., Hodges, S., Freeman, D., Davison, A. and Fitzgibbon, A., 2011. KinectFusion: Real-Time 3D Reconstruction and Interaction using a Moving Depth Camera. UIST '11 Proceedings of the 24th annual ACM symposium on User Interface Software and Technology, pp. 559-568.

Wikipedia: Kinect, available online (accessed 2012): http://en.wikipedia.org/wiki/Kinect

Lowe, D. G., 2004. Distinctive image features from scale-invariant keypoints. International journal of computer vision, 60(2), pp. 91-110.

Newcombe, R.A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A.J., Kohli, P., Shotton, J., Hodges, S. and Fitzgibbon, A., 2011. KinectFusion: Real-Time Dense Surface

Mapping and Tracking. ISMAR '11 Proceedings of the 2011 10th IEEE International Symposium on Mixed and Augmented Reality, pp. 127-136.

Point Cloud Library (PCL), available online (accessed 2012): http://pointclouds.org

Wenzel, K., Abdel-Wahab, M., Cefalu, A. and Fritsch, D., 2011. A Multi-Camera System for Efficient Point Cloud Recording in Close Range Applications. Low-Cost 3D (LC3D) workshop, Berlin, December 2011.