F O U N D A T I O N S O F C O M P U T I N G A N D D E C I S I O N S C I E N C E S
Vol. 42 (2017) No. 3
ISSN 0867-6356
DE GRUYTER OPEN DOI: 10.1515/fcds-2017-0009 e-ISSN 2300-3405

MODREG: A MODULAR FRAMEWORK FOR RGB-D IMAGE ACQUISITION AND 3D OBJECT MODEL REGISTRATION

Tomasz KORNUTA *†, Maciej STEFAŃCZYK †

Abstract. RGB-D sensors became a standard in robotic applications requiring object recognition, such as object grasping and manipulation. A typical object recognition system relies on matching of features extracted from RGB-D images retrieved from the robot sensors with the features of the object models. In this paper we present ModReg: a system for registration of 3D models of objects. The system consists of a modular software associated with a multi-camera setup supplemented with an additional pattern projector, used for the registration of high-resolution RGB-D images. The objects are placed on a fiducial board with two dot patterns enabling extraction of masks of the placed objects and estimation of their initial poses. The acquired dense point clouds constituting subsequent object views undergo pairwise registration and at the end are optimized with a graph-based technique derived from SLAM. The combination of all those elements resulted in a system able to generate consistent 3D models of objects.

Keywords: ModReg, textured stereo, RGB-D image, point cloud, registration

1. Motivation of the work

Emergence of new types of sensors stimulates robotics. Up to the late nineties the most popular sensors used for obstacle avoidance were sonars, whereas the commercialization of laser rangefinders (such as the Hokuyo UTM and SICK LMS series) enabled rapid development of methods for 2D robot pose estimation [21] or techniques for simultaneous localization and map building (SLAM) [38]. Similarly, the appearance of inexpensive RGB-D sensors [33, 15], such as Microsoft Kinect released

*IBM Research, Almaden, 650 Harry Rd, San Jose, CA 95120, tkornuta@gmail.com
†Warsaw University of Technology, Institute of Control and Computation Engineering, stefanczyk.maciek@gmail.com

in 2010, triggered a massive wave of interest in 3D visual perception [28]. Access to the color images supplemented with depth maps provided by such sensors changed the the entire field of robotics, starting with their application to map building [26] and SLAM [36, 3], through recognition of human posture [7] or hand gestures [27], ending on object grasping [6].

A typical perception subsystem for object recognition (e.g. [14]) relies on matching between features of the object model and features extracted from the image of the scene (e.g. Hough voting [39]), followed by correspondence grouping and hypothesis verification (e.g. [1]). This requires the system to possess the three-dimensional models of objects, whereas the most convenient form of such models are dense point clouds supplemented with sparse clouds of features (e.g. [2]). Those features can be extracted from color images and then projected into Cartesian space on the basis of the associated depth maps (e.g. [37]), from point clouds (e.g. [29, 40]) or extracted straight from RGB-D images (multi-modal features [2]). Generation of grasps [31], robust estimation of object poses in the bin picking task [10] or robotic cutting of fruits and vegetables [19] are only selected exemplary robotic systems relying on such models. Despite the existence of databases full of 3D CAD models (such as 3D Warehouse[1]), the models frequently lack important cues (e.g. texture), are rendered without a realistic lighting etc., thus are called non-photorealistic models [25]. Besides, due to their multiplicity, it is often impossible to find exact models of the currently surrounding us objects, such as cups, bowls, bottles, vegetables, fruits, i.e. we lack the models of the objects of everyday use that the service robots are supposed to grasp and manipulate. Thus a problem of acquisition of 3D models on the basis of images of physical objects having diverse shapes, colors and textures appears. The introduced ModReg system facilitates this process.

1.1. Contributions of the paper

The following paper is an extension of a paper [16]. Besides, some of the preliminary results of our work in this subject were presented in [17], which introduced the novel, multi-camera setup with a texture projector used in this research. Finally, we have also used ModReg to capture a significant part of the WUT Visual Perception Dataset described in [35].

This paper summarizes our efforts in registration of models of objects and its main contributions are twofold. First, we provide a comprehensive description of the ModReg system that was partially presented in the papers mentioned above. Second, we present the results of its quantitative verification, consisting of analysis of results achieved by individual components (e.g. pairwise point cloud registration, loop closure) as well as overall performance of the system using the multi-camera setup when compared with MS Kinect.

[1] `http://3dwarehouse.sketchup.com`

1.2. Structure of the paper

The paper is structured as follows. In the next section we discuss the problems appearing during the registration of models of objects. Next we present the structure of the ModReg system, followed by the description of the multi-camera system supplemented with an additional pattern projector and main software modules of the system. For completeness, as we treated the results obtained with the use of the Kinect sensor as reference, we also present the Kinect acquisition module. We conclude the paper with the analysis of the achieved results.

2. Registration of the models of objects

The problem of combination of multiple views of the scene into a single, consistent 3-D model is known as registration [12, 26]. When dealing with models generated with the use of RGB-D sensors the problem of registration can be reformulated as finding the relative positions and orientations of the separately acquired point clouds followed by the transformation of coordinates of all points into a common coordinate frame. The principle of operation of such a typical registration system is as follows. After the acquisition of a pair of point clouds an initial transformation between them is estimated. For this purpose two sets of features must be extracted from both clouds. Those features are matched together in order to find correspondences, used subsequently for finding the initial estimation of the transformation. This estimate can then be refined, typically on the basis of correspondences between points of the original, dense clouds. Because those operations are made on pairs of point clouds, this approach is known under the name pairwise registration. Finally, in order to reduce the model inconsistencies, a global optimization may be applied. Systems working according to this scheme can be also utilized for generation of models of particular objects of everyday use.

There are several existing software solutions (both commercial such as Microsoft's 3D Scan or Agisoft's PhotoScan, and open source, e.g. KinectFusion [24] in PCL [30]) able to build consistent models of objects from a set of their images. Sadly, they typically return mesh-based models, whereas during the recognition we require the models based on point clouds along with the associated features extracted from color/depth. This motivated us to develop our own model system. However, after we have integrated our software it appeared that the generated models were far from ideal, there were many discrepancies, the number of model features was quite low etc. The most important of the identified problems are as follows.

First, the commercial RGB-D sensors relying on structured light technique (such as Microsoft Kinect or Orbbec Astra Pro) can properly estimate depth starting from around 0.4m, thus a typical point cloud of an object of everyday use acquired from such a distance will contain few thousand points at the most. This affects both the proper estimation of the initial transition between subsequent views and the overall low resolution and quality of the model. The initial transformation can be estimated by introduction of appropriate visual markers (e.g. by placing the object on a board

with known, easy to recognize patterns [2]). Such an estimation is usually a good starting point for further refinement. Unfortunately, this requires to observe the object from even further distance (as the patterns must also be visible), thus the problem of the low quality of the final model remains.

Another problem concerns distinction between the points belonging to the object and the points constituting the background. Some data sets (e.g. Washington RGB-D Object Dataset [18]) provide such an information in the form of an additional binary image containing the mask of the object. It is required that a registration system should provide tools solving this problem as well.

Finally, it is crucial that the generated model should be extremely consistent. An error of even few centimeters is negligible in the case of registration of outdoor scenes or buildings, whereas in the case of everyday objects it will result in not only slightly inconsistent models, but foremost will affect the accuracy of the pose estimation during object recognition phase and may trigger serious problems during object grasping. Thus, the use of global optimization methods based on loop closure seems necessary, and once again indicates the necessity for high quality input images.

After the analysis of the aforementioned problems we have revised all the elements of the registration procedure, from data acquisition hardware to the global optimization, which resulted in the system presented in the next section.

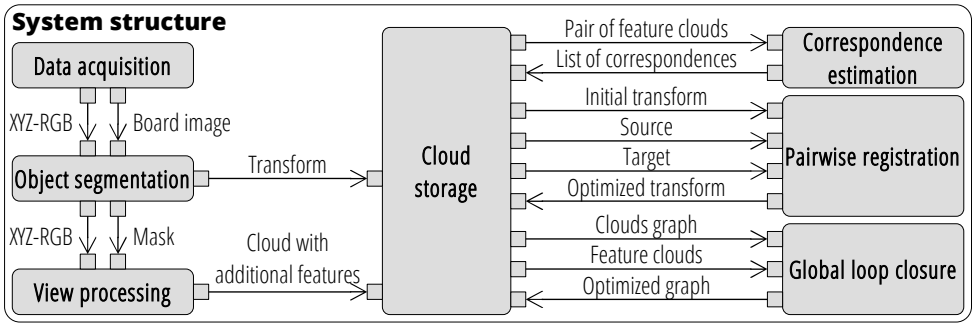


Figure 1: General system structure

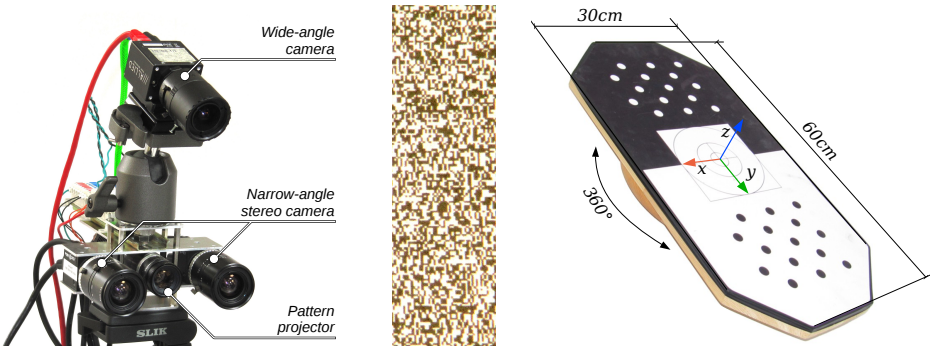
3. The ModReg system

The general structure of the ModReg (Modular Registration) system is presented in fig. 1. The system is modular and follows the blackboard architecture, with Cloud storage being the central module, governing the actions and collecting results from the associated modules. Those modules were implemented as components of DisCODe [34] framework, encapsulating, besides others, classes and functions from two computer vision libraries: OpenCV [5] and PCL [30]. All those modules are presented

in the following subsections. Both the DisCODe framework[2] and ModReg system[3] are distributed under the MIT License and their source codes are publicly available on-line.

3.1. Data acquisition

Data acquisition module is responsible for gathering of images containing different sides of the object of interest. Because we wanted the system to be able to work with different hardware configurations, this module is also responsible for standardization of the format of the acquired view (sensor readings). We decided that such a view will consist of an organized point cloud in the form of RGB image supplemented with XYZ image (image containing Cartesian coordinates of each pixel), along with the image containing a fiducial board (please refer to fig. 3).



(a) (b) (c)

Figure 2: Key components of the acquisition system: (a) The multi-camera setup with an additional texture projector, (b) close-up of the projected pattern, (c) turntable with two dot patterns

3.1.1. Turntable

We use a fiducial board with circular patterns (fig. 2c), similarly as it was done by Willow Garage in Object Recognition Kitchen (ORK). Two dot patterns are used for determination of the board pose with respect to the sensor frame regardless of occlusions (however, it is required that at least one pattern is fully visible in every view). The axis of rotation of the board is aligned with z axis of the coordinate frame. During the view acquisition the object is placed at the center of the board. This enables relatively accurate initial pose estimation and fosters the transformation

[2] `https://github.com/DisCODe/DisCODe`
[3] `https://github.com/DisCODe/Registration`

of all acquired point clouds into a common, board-centered coordinate frame, as well as will facilitate further object segmentation.

3.1.2. Textured stereovision

The main hardware setup (fig. 2a) of ModReg consists of three cameras supplemented with an additional texture projector. The first two cameras, namely PointGrey BFLY-PGE-13S2C-CS color cameras with long-focal lenses (narrow, 33° horizontal field of view), formed a stereo pair, with a 10 cm baseline. This enabled to register an object of size 15 cm from the distance of 70 cm, and the object area occupies almost the whole image, having typically few hundred thousand pixels per object view.

For the generation of disparity maps we have used the classical Semi-Global Block Matching (SGBM) stereovision algorithm [11]. As we wanted the acquired depth maps to be as dense as possible we have used an EFFI-Lase LED projector with a pseudo-random dots pattern (fig. 2b). Such a hardware configuration can be classified as textured stereo [13, 33, 15].

The projected white pattern influences the perceived color image of the object. For this reason we collect two pairs of images for each view – first pair with projector turned on and the second while illuminating the object only with a diffused light (fig. 3). The former is used for generation of a dense point cloud of the object, whereas the latter is used to color the cloud and to extract keypoints with descriptors used in subsequent processing steps.

Because our goal was to acquire as many object points as possible, we wanted the field of view of the stereo pair to be the narrow. That contradicted capturing the markers of the fiducial board, needed for initial estimation of the pose of the object. For this reason we incorporated the third camera, Prosilica GC1290C with 56° lens, mounted above and slightly tilted to the stereo pair. Such a configuration enabled us to capture the whole board and subsequently determine its pose. Please notice that despite the actual number of used cameras for the simplicity throughout the rest of this paper we will refer to this setup as to stereo – as during computation of depth maps/point clouds we in fact rely only on the information from the stereo pair.

The data flow between the components responsible for the generation of a single

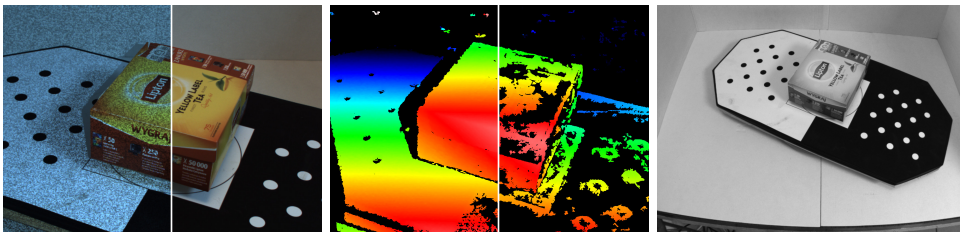


Figure 3: (From left) Color images registered with projected pattern and diffused light, depth registered with projected pattern and diffused light, view of the same scene from the wide-angle camera

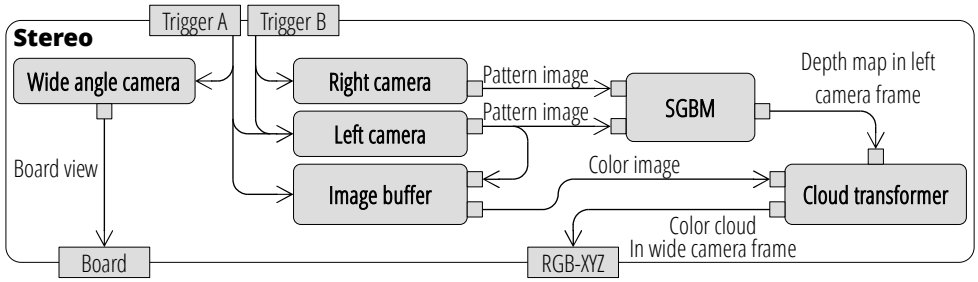


Figure 4: Activity diagram for acquisition from textured multi-camera setup

object view is presented in fig. 4. Two triggers are responsible for acquisition of board view and color image from left camera (trigger A) and acquisition of depth map from stereo cameras (trigger B). Those external signals synchronize cameras, as well as switch two illumination sources back and forth between pattern projector and diffused light.

3.1.3. Structured light

Second hardware setup consists of a single Microsoft Kinect sensor. The size of the board along with rather wide horizontal angle of sensor view (62°) and lower limit of depth acquisition of the sensor enabled us for using the same RGB image for both detection of the board and generation of point cloud, as presented in fig. 5.

Utilization of a structured light projector enables acquisition of a dense depth map disregarding the lack of edges, object texture etc. However, the low resolution of the acquired depth image results in a quite small number of points (typically several thousands) constituting the object (fig. 6).

3.2. Object segmentation

The Object segmentation module is responsible for two tasks: transformation of the coordinates from sensor to board reference frame and for the generation of the mask distinguishing the pixels constituting the object from the background. The activity

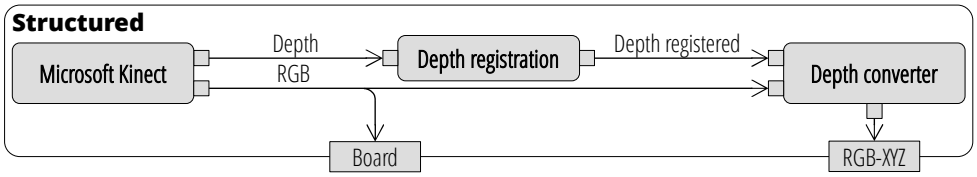


Figure 5: Activity diagram for structured light image acquisition

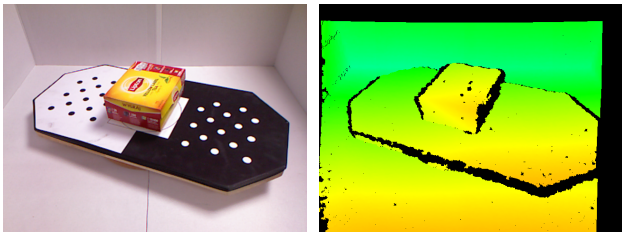


Figure 6: Image registered with Microsoft Kinect (left – color image, right – depth map, black means no data)

diagram of this module is presented in fig. 7.

The board pose is estimated by recognizing two patterns (white dots on black background and black dots on white) and passing the dots centers to the PnP (Perspective-n-Point) problem solver. This transformation is then used for transformation of the cloud from the camera to board frame, which makes the axes of coordinate system aligned to the object. At this point the object mask can be generated by applying the axis-aligned bounding box (3D cuboid) to processed data (e.g. $x, y \in [-10cm, 10cm]$ and $z \in [1cm, 10cm]$). As it was mentioned earlier, views obtained from the Kinect sensor contain much lower number of points when compared to multi-camera setup. Exemplary results for both hardware setups are presented in fig. fig. 8. In this case the Kinect mask contained around 15k points, where for stereo it was around 218k. We obtained similar ratios for all the objects and views.

3.3. View processing

The View processing module is responsible for two major tasks: filtration of the input cloud in order to remove points not belonging to the mask and extraction of object features. In the current implementation we have decided to use SIFT (Scale Invariant Feature Transform) features [20] extracted from RGB image, as they still appear to be one of the most robust photogrammetric features [8]. The features are next transformed with the use of Cartesian coordinates from XYZ image into sparse cloud of features. The features, represented as intensive red dots, can be observed e.g. in

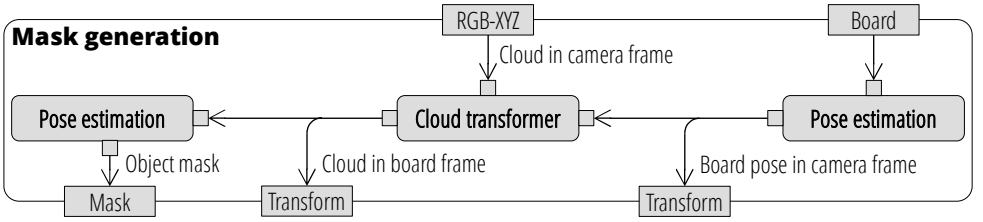


Figure 7: Activity diagram of the object segmentation module

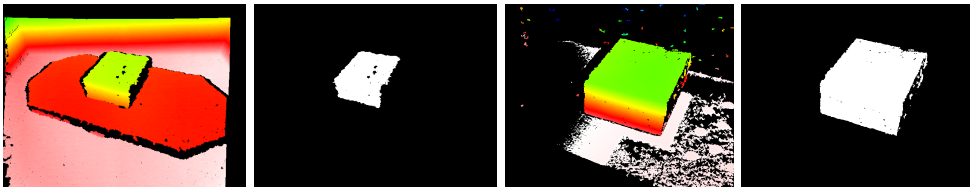


Figure 8: Depth transformed to centre of the board frame and generated object masks: (left) from Kinect sensor (right) from multi-camera setup

fig. 11 and fig. 12.

Additionally, the point clouds constituting the object undergo filtering in order to reject invalid pixels resulting from the noise introduced by the used sensors, depth estimation algorithms etc. This is currently realized with the use statistical outlier filter, which removes all points being inconsistent with local statistics (a Gaussian distribution is assumed).

3.4. Cloud storage

Cloud storage is the central module of the system, responsible for storing the point clouds and enabling the user to decide (with the use of GUI) what and when should be passed to the associated modules.

Based only on data generated by the previous modules the first approximation of the object model can be created. Relying on the initial transformation estimations (from the dot patterns) for alignment of the clouds results in small, yet visible inconsistencies of the model, as presented in fig. 9.

3.5. Pairwise registration

The Pairwise registration module is responsible for aligning of the pairs of point clouds with the use of the Iterative Closest Point (ICP) [4]. Fig. 10 presents a general

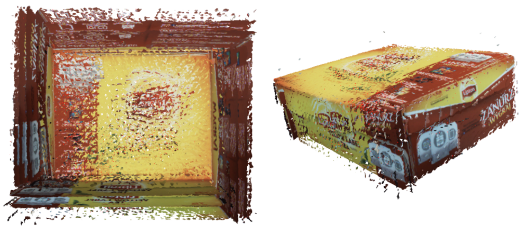


Figure 9: Object model aligned relying on the initial estimations. Notice splits of the box sides visible in the bottom view (left)

activity diagram realizing the ICP-based pairwise registration of point clouds. The module inputs consist of the source (i.e. reference) cloud and the target cloud (the one which transformation will be optimized), along with the initial transformation (i.e. calculated on the basis of the estimation of the pose of the fiducial board) from the Cloud storage. The module outputs the refined transformation (or a set of transformations when inputs contain a set of clouds with set of initial transformations between them).

Currently the user can switch between several ICP flavours, whereas the best results were obtained with ICP using two-step data association relying on normal vectors and color [22]. Representative results, obtained for the box object, are presented in table 1.

Table 1: Convergences of different types of ICP for the sequence containing a box object

ICP flavour	**Not Converged** [%]	**Bad Convergence** [%]	**Success rate** [%]	**Mean number of iterations** [-]
Standard	0	51.15	48.85	27.12
Normals	6.39	4.99	88.62	10.84
color	0	11.25	88.75	21.92
color + normals	1.92	6.01	92.07	11.16

3.6. Correspondence estimation

This module is responsible for finding the correspondences between sparse clouds of features associated with the two analyzed dense point clouds. The reciprocal correspondences are estimated with FLANN (Fast Library for Approximate Nearest Neighbors) [23], using kd-trees for fast computation of the similarity of feature descriptors.

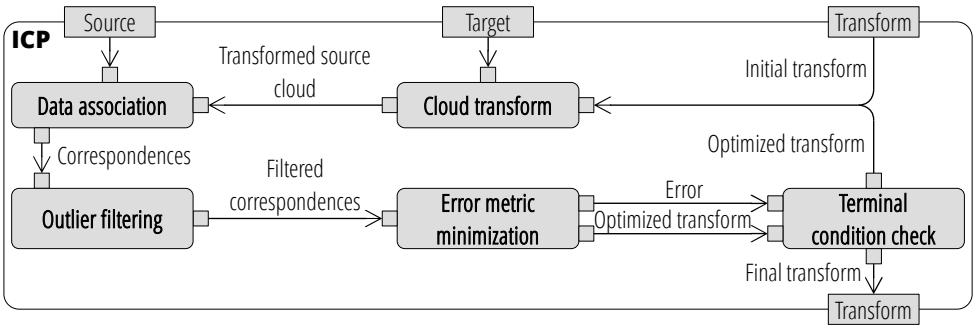


Figure 10: General structure of the ICP algorithm for pairwise cloud registration

Additionally, the module enables rejection of some of the correspondences. We tested two rejection methods: a simple method relying on distance between Cartesian positions of the corresponding keypoints and a RANSAC-based method for rejecting the outliers that do no match the hypothesis regarding the transformation between two clouds. As the correspondences are found between two quite well aligned point clouds, the simpler method gave comparable results, with additionally having a quite important advantage: stability – which is the drawback of RANSAC-based correspondence filtering in the case of small number of input set of correspondences (fig. 11).

3.7. Loop closure

The Loop closure module is responsible for performing the graph based optimization, where the graph vertices represent point clouds constituting the consecutive views, whereas edges represent the found correspondences. In the current version of the system we utilize the LUM algorithm [21], which estimates the costs of a given edge as the sum of Euclidean distance between Cartesian coordinates of the corresponding keypoints. A comparison of models generated with and without the loop closure is presented in fig. 12. Another considered solution was the Explicit Loop Closing

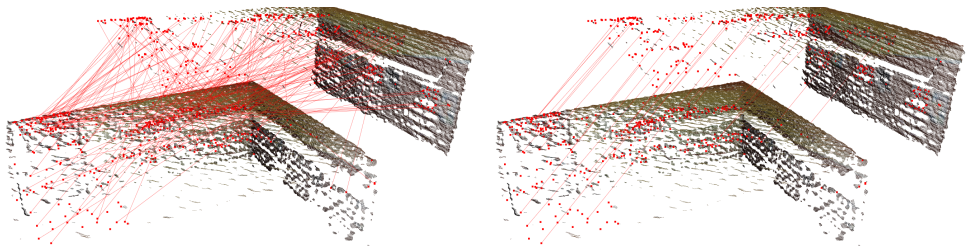


Figure 11: Correspondences between two consecutive views: estimated with FLANN (left) and remaining after simple Cartesian distance rejection (right). The clouds were manually shifted only for the visualization purposes

Table 2: Comparison of results obtained for two selected loop closure algorithms

Translation noise					**Rotation noise**				
Input	**LUM error**		**ELCH error**		**Input**	**LUM error**		**ELCH error**	
[mm]	[mm]	[°]	[mm]	[°]	[°]	[mm]	[°]	[mm]	[°]
0	0.03	0.25	0.11	0.02	5	0.00	0.09	7.32	6.96
5	0.02	0.07	4.15	0.12	10	0.00	0.19	10.40	11.95
10	0.02	0.11	2.08	2.67	30	0.03	0.09	4.54	29.31
50	0.02	0.17	30.00	5.54	60	0.00	0.10	66.04	71.82
100	0.05	0.29	66.67	9.34	90	46.80	255.43	95.34	113.52

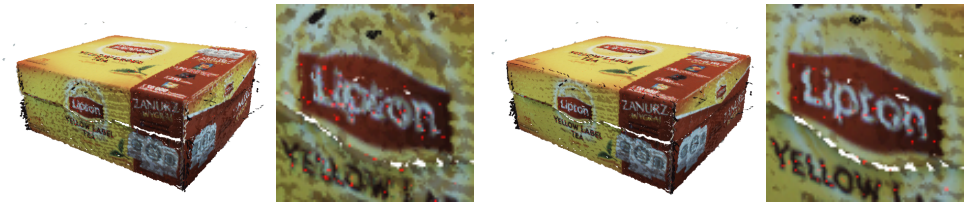


Figure 12: Object model after pairwise registration (left) and optimized with the LUM-based global loop closure (right). The improvement can be seen e.g. in the Lipton brand logo

Heuristic algorithm (ELCH) [32], but the errors obtained during building models of small objects in the case of controlled (input) transformations between point clouds constituting the object were much worse (please refer to table 2).

4. Analysis of results

We have performed several sets of experiments on different objects in order to validate the ModReg system. The goal of the first set of experiments was to quantitatively compare the numbers of object points acquired with both hardware configurations. For this reason we have acquired views of three objects possessing highly different shapes: a cuboidal Lipton tea, a cylindrical Inka coffee can and a Sugar bag having

Object	**Kinect Sensor**		**Textured Stereo**		**Ratio** [-]
	Exemplary view	**Avg. number of points** [-]	**Exemplary view**	**Avg. number of points** [-]	
Sugar bag		7250.1		129342.4	17.8
Lipton tea		9320.3		135702.8	14.6
Inka coffee		5597.7		162623.3	29.1

Table 3: Comparison of average number of points constituting selected objects for all object views

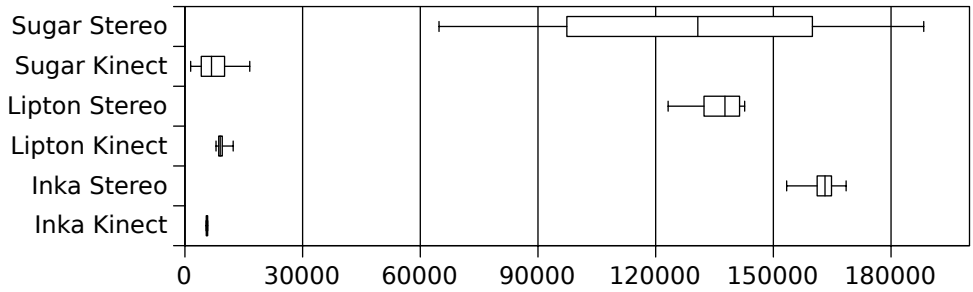


Figure 13: Statistics (means, medians and standard deviations) of number of points constituting given object in views acquired with both Kinect sensor and Textured Stereo

an irregular shape. As presented in table 3, the low resolution of Kinect depth images resulted in a quite small object point clouds, containing typically several thousand of points, whereas the view of the same scene acquired with our multi-camera system resulted in the object mask containing significantly more points. The last column, containing the ratios between the numbers of points constituting object views acquired from stereo and acquired from the Kinect sensor, shows that the former views were larger typically by at least an order of magnitude (a factor of 20 in average). Table 3 provided only mean numbers of points of views of the selected objects. A more detailed statistics regarding the number of acquired object points, including ranges (i.e. maximum and minimum number of points) along with means and standard deviations, are presented in fig. 13. The former presents the statistics of object points acquired with the use of Kinect sensor, whereas the latter shows the results for both hardware setups.

The goal of the next set of experiments was to verify and compare results of the whole object registration procedure when applied to data acquired from both hardware configurations. The obvious method would require utilization of an external device providing some kind of a ground-truth. This is a typical case when it comes to datasets for object recognition, pose estimation or semantic scene analysis [9]. However, as we did not have such a device and there are no models (e.g. CAD models) of the registered objects available (thus our motivation to register such models), we had to propose a new verification method.

The idea of the method relies on the comparison of the registered model of an object with a manually generated solid model of that object using the known object shape and dimensions. This enables projection of both models into the same reference frame and, finally, calculation of the registration errors by calculation of distances between points constituting the registered model with the surface of the manually generated model. We have selected two objects with simple shapes: a cuboid (Lipton tea box) and a cylinder (Inka coffee can) of known dimensions. The results, comprising of RMSE and maximum distance of registered model points from the generated solid model surface, are presented in table 4. In the case of cuboidal Lipton, dimensions

include width, depth and height, whereas for cylindrical Inka those include diameter of the cylinder and its height. The results achieved on the basis of views acquired by our textured stereo system are clearly better, with both RMSE and maximum errors reduced by half.

Object	**Dimensions**	**Kinect Sensor**		**Textured Stereo**	
		RMSE	**Max error**	**RMSE**	**Max error**
	[mm]	[mm]	[mm]	[mm]	[mm]
Lipton tea	150x125x60	0.91	1.9	0.5	0.98
Inka coffee	98x150	1.09	5.6	0.61	2.5

Table 4: Comparison of errors of selected models of objects registered relying on data acquired from both hardware configurations

The examples of the models of objects registered with the use of our ModReg system are presented in fig. 14. We have achieved a quite consistent models of both regular (such as tea boxes) and irregular objects (e.g. bag of sugar). It can be observed that the combination of ICP with LUM managed to successfully remove inconsistencies of the resulting model (like the blurred texts and logos visible in columns 2 and 3). Besides, in the case of highly irregular objects, such as partially filled bag of sugar, all improvements applied to clouds acquired from Kinect appeared to fail (row 4). This resulted from the low resolution and high deformations of surfaces, thus both optimizations failed to find good correspondences, both in terms of color points (ICP) and features (LUM). In contrast, the multi-camera system successfully generated a highly consistent models of the object.

5. Summary

In this paper we have presented a modular system for registration of 3D models of objects called ModReg. The modularity of the software enabled to use and compare two sources of RGB-D images, i.e. Kinect sensor and a novel multi-camera setup supplemented by an additional pattern projector The latter setup was used for the registration of high-resolution RGB-D images, which in turn resulted in better (i.e. having higher resolution and being more consistent) models of objects. The presented experiments confirm the usefulness of the developed solution.

A major drawback of the system is that due to the photometric nature of the used features its application is limited to textured objects. This problem might be overcome by aggregation of point clouds into more abstract structures (such a lines, surfaces or meshes) and using more sophisticated registration algorithms, such as e.g. ICP flavours relying on lines (ICL). In the future we also plan to extend the system by introducing other types of features, extracted from both color and depth, in order to make the system more robust.

Besides, we are also working on incorporation of new types of RGB-D sensors, that

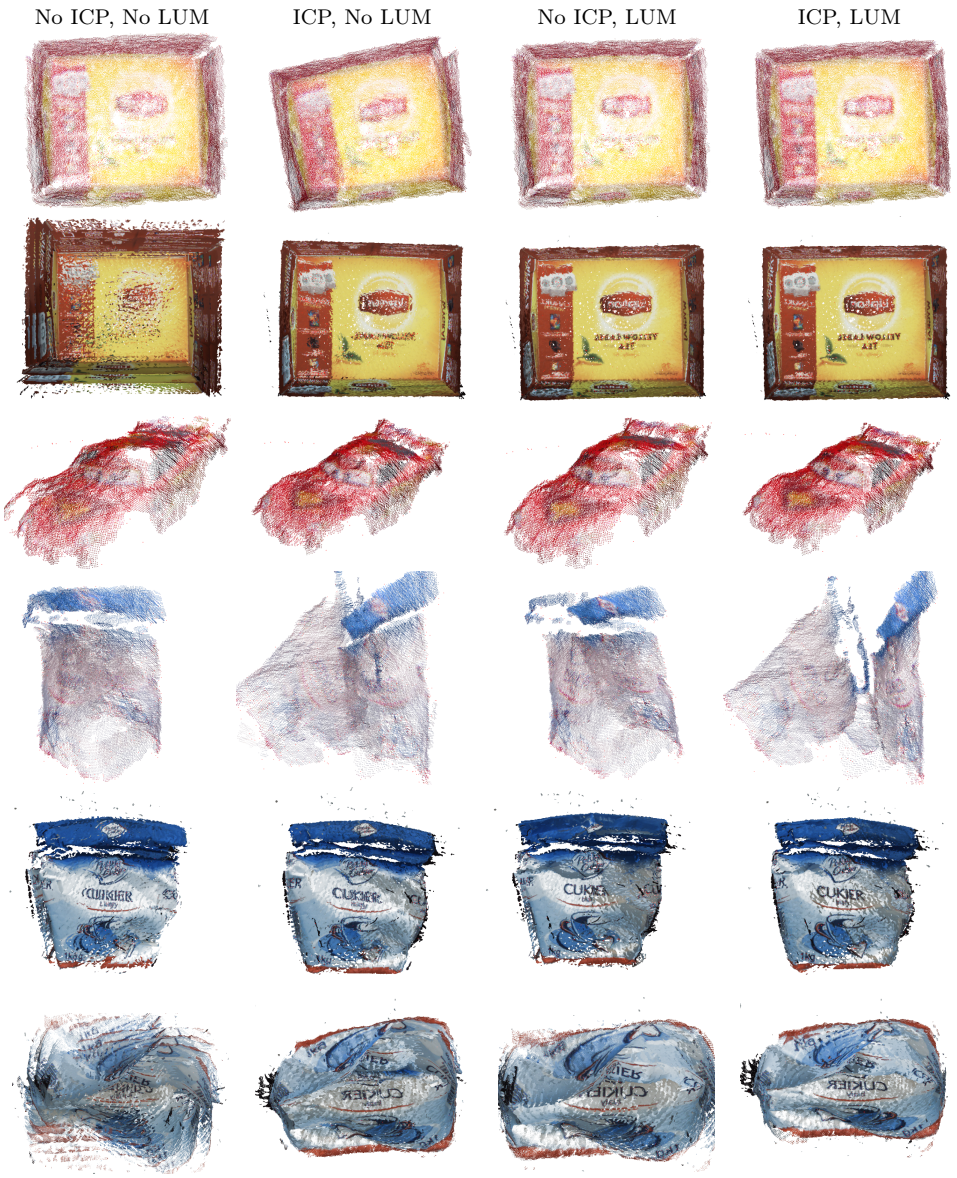


Figure 14: Exemplary results of registration. Columns contain (from left): registration on the basis of board pose estimation, with the use of ICP with color, without ICP but with the use of LUM, using both ICP color and LUM. Rows contain (from top): a Lipton tea box (data from Kinect, bottom views), the same Lipton tea box (from textured stereo, bottom views), a Lightning McQueen toy (from Kinect, side views), a sugar bag (from Kinect, side views) and the same sugar bag (textured stereo, side and bottom views)

recently appeared on the market (e.g. Intel Realsense R200). In particular, utilization of Kinect XBO will enable us to compare three most popular depth acquisition methods. The modularity of the ModReg system will highly facilitate those tasks. Following the community standards we have made the source code of the ModReg system to be publicly available[4].

Finally, the developed system was used for capturing of the significant amount of data constituting the WUT Visual Perception Dataset [35], the dataset created for the purpose of development, comparison and evaluation of diverse algorithms for object model registration and object recognition. The dataset is also publicly available on-line[5].

Acknowledgment

The authors would like to thank Marta Łępicka and Mikołaj Kamionka for help with the evaluation of selected registration algorithms and kindly acknowledge the support of the National Science Centre according to the decision number DEC-2012/05/D/ST6/03097, the National Centre for Research and Development grant no. PBS1/A3/8/2012 and grant of the Dean of Faculty of Electronics and Information Technology of Warsaw University of Technology no. 504/01446/1031/42.

References

[1] Aldoma A., Tombari F., Di Stefano L., and Vincze M. A global hypotheses verification method for 3D object recognition. In *Computer Vision (ECCV 2012)*, pages 511–524. Springer, 2012.

[2] Aldoma A., Tombari F., Prankl J., Richtsfeld A., Di Stefano L., and Vincze M. Multimodal cue integration through hypotheses verification for RGB-D object recognition and 6DOF pose estimation. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 2104–2111. IEEE, 2013.

[3] Belter D., Nowicki M., Skrzypczyński P., Walas K., and Wietrzykowski J. Lightweight RGB-D SLAM System for Search and Rescue Robots. In *Progress in Automation, Robotics and Measuring Techniques*, pages 11–21. Springer, 2015.

[4] Besl P. and McKay N. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239 –256, 1992.

[5] Bradski G. and Kaehler A. *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly Media, 2008.

[6] Correll N., Bekris K. E., Berenson D., Brock O., Causo A., Hauser K., Okada K., Rodriguez A., Romano J. M., and Wurman P. R. Analysis and observations from

[4] `https://github.com/DisCODe/Registration`
[5] `http://robotyka.ia.pw.edu.pl/datasets/`

the first amazon picking challenge. *IEEE Transactions on Automation Science and Engineering*, 2016.

[7] Dziergwa M., Kaczmarek P., and Kędzierski J. RGB-D Sensors in Social Robotics. *Journal of Automation Mobile Robotics and Intelligent Systems*, 9(1):18–27, 2015.

[8] Figat J., Kornuta T., and Kasprzak W. Performance evaluation of binary descriptors of local features. In Chmielewski L., Kozera R., Shin B.-S., and Wojciechowski K., editors, *Proceedings of the International Conference on Computer Vision and Graphics*, volume 8671 of *Lecture Notes in Computer Science*, pages 187–194. Springer Berlin / Heidelberg, 2014.

[9] Firman M. Rgbd datasets: Past, present and future. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 19–31, 2016.

[10] Großmann B., Siam M., and Krüger V. Comparative evaluation of 3D pose estimation of industrial objects in RGB pointclouds. In *Computer Vision Systems*, pages 329–342. Springer, 2015.

[11] Hirschmuller H. Stereo processing by semiglobal matching and mutual information. *IEEE Trans. Pattern Anal. Mach. Intell.*, pages 328–341, II 2008.

[12] Holz D., Ichim A. E., Tombari F., Rusu R. B., and Behnke S. Registration with the Point Cloud Library: A Modular Framework for Aligning in 3-D. *Robotics & Automation Magazine, IEEE*, 22(4):110–124, 2015.

[13] Konolige K. Projected texture stereo. In *International Conference on Robotics and Automation (ICRA)*, pages 148–155. IEEE, 2010.

[14] Kornuta T. and Laszkowski M. Perception subsystem for object recognition and pose estimation in RGB-D images. In Szewczyk R., Zieliński C., and Kaliczyńska M., editors, *Recent Advances in Automation, Robotics and Measuring Techniques*, volume 440 of *Advances in Intelligent Systems and Computing (AISC)*, pages 597–607. Springer, 2016.

[15] Kornuta T. and Stefańczyk M. Acquisition of RGB-D images: sensors (in Polish). *Pomiary – Automatyka – Robotyka PAR*, 18(2):92–99, 2014.

[16] Kornuta T. and Stefańczyk M. Comparison of methods of aquisition of RGB-D images for the purpose of registration of three-dimensional models of objects (in Polish). In *XIV Krajowa Konferencja Robotyki – Postępy robotyki*, volume 2, pages 357–366, 2016.

[17] Kornuta T. and Stefańczyk M. Utilization of textured stereovision for registration of 3D models of objects. In *21th IEEE International Conference on Methods and Models in Automation and Robotics, MMAR'2016*, pages 1088–10093. IEEE, 2016.

[18] Lai K., Bo L., Ren X., and Fox D. A large-scale hierarchical multi-view RGB-D object dataset. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 1817–1824. IEEE, 2011.

[19] Lenz I., Knepper R., and Saxena A. DeepMPC: Learning deep latent features for model predictive control. In *Proceedings of Robotics: Science and Systems*, Rome, Italy, July 2015.

[20] Lowe D. Object recognition from local scale-invariant features. In *Computer Vision, The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1150–1157. Ieee, 1999.

[21] Lu F. and Milios E. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4(4):333–349, 1997.

[22] Łępicka M., Kornuta T., and Stefańczyk M. Utilization of colour in ICP-based point cloud registration. In *Proceedings of the 9th International Conference on Computer Recognition Systems (CORES 2015)*, volume 403 of *Advances in Intelligent Systems and Computing*, pages 821–830. Springer, 2016.

[23] Muja M. and Lowe D. G. Fast approximate nearest neighbors with automatic algorithm configuration. In *VISAPP (1)*, pages 331–340, 2009.

[24] Newcombe A. J., Richard A and Davison, Izadi S., Kohli P., Hilliges O., Shotton J., Molyneaux D., Hodges S., Kim D., and Fitzgibbon A. KinectFusion: Real-time dense surface mapping and tracking. In *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*, pages 127–136. IEEE, 2011.

[25] Peng X., Sun B., Ali K., and Saenko K. Learning deep object detectors from 3d models. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1278–1286, 2015.

[26] Pomerleau F., Colas F., and Siegwart R. A review of point cloud registration algorithms for mobile robotics. *Foundations and Trends in Robotics (FnTROB)*, 4(1):1–104, 2015.

[27] Ramey A., González-Pacheco V., and Salichs M. A. Integration of a low-cost RGB-D sensor in a social robot for gesture recognition. In *Proceedings of the 6th international conference on Human-robot interaction*, pages 229–230. ACM, 2011.

[28] Ren X., Fox D., and Konolige K. Change Their Perception: RGB-D for 3-D Modeling and Recognition. *Robotics & Automation Magazine, IEEE*, 20(4):49–59, 2013.

[29] Rusu R., Blodow N., and Beetz M. Fast point feature histograms (FPFH) for 3D registration. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 3212–3217. IEEE, 2009.

[30] Rusu R. B. and Cousins S. 3D is here: Point Cloud Library (PCL). In *International Conference on Robotics and Automation*, Shanghai, China, 2011 2011.

[31] Seredyński D. and Szynkiewicz W. Fast grasp learning for novel objects. In *Recent Advances in Automation, Robotics and Measuring Techniques*, volume 440 of *Advances in Intelligent Systems and Computing (AISC)*, pages 681–692. Springer, 2016.

[32] Sprickerhof J., Nüchter A., Lingemann K., and Hertzberg J. A heuristic loop closing technique for large-scale 6D SLAM. *Automatika: Journal for Control, Measurement, Electronics, Computing & Communications*, 52(3), 2011.

[33] Stefańczyk M. and Kornuta T. Acquisition of RGB-D images: methods (in Polish). *Pomiary – Automatyka – Robotyka PAR*, 18(1):82–90, 2014.

[34] Stefańczyk M. and Kornuta T. Handling of asynchronous data flow in robot perception subsystems. In *Simulation, Modeling, and Programming for Autonomous Robots*, volume 8810 of *Lecture Notes in Computer Science*, pages 509–520. Springer, 2014.

[35] Stefańczyk M., Laszkowski M., and Kornuta T. WUT Visual Perception Dataset – a dataset for registration and recognition of objects. In *Challenges in Automation, Robotics and Measurement Techniques*, volume 440 of *Advances in Intelligent Systems and Computing (AISC)*, pages 635–645. Springer, 2016.

[36] Sturm J., Engelhard N., Endres F., Burgard W., and Cremers D. A benchmark for the evaluation of RGB-D SLAM systems. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 573–580. IEEE, 2012.

[37] Tang J., Miller S., Singh A., and Abbeel P. A textured object recognition pipeline for color and depth image data. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 3467–3474. IEEE, 2012.

[38] Thrun S., Burgard W., and Fox D. A probabilistic approach to concurrent mapping and localization for mobile robots. *Autonomous Robots*, 5(3-4):253–271, 1998.

[39] Tombari F. and Di Stefano L. Object recognition in 3D scenes with occlusions and clutter by hough voting. In *Image and Video Technology (PSIVT), 2010 Fourth Pacific-Rim Symposium on*, pages 349–355. IEEE, 2010.

[40] Tombari F., Salti S., and Di Stefano L. Unique signatures of histograms for local surface description. In *Computer Vision–ECCV 2010*, pages 356–369. Springer, 2010.

This is an extended version of the paper presented at the 14th National Conference on Robotics (KKR 2016), Polanica Zdrój, Poland, September 14-18, 2016.