A TOOL FOR CROWDSOURCED BUILDING INFORMATION MODELING THROUGH LOW-COST RANGE CAMERA: PRELIMINARY DEMONSTRATION AND POTENTIAL

F. Capocchiano [a, b], R. Ravanelli [a,*] M. Crespi [a, b]

[a] Geodesy and Geomatics Division, DICEA - University of Rome "La Sapienza", Rome, Italy
[b] Sapienza School for Advanced Studies, Rome, Italy
<roberta.ravanelli, mattia.crespi>@uniroma1.it

Commission II

KEY WORDS: Building Information Modeling, scan to BIM, range camera, Occipital Structure Sensor[TM], algorithm, crowdsourcing

ABSTRACT:

Within the construction sector, Building Information Models (BIMs) are more and more used thanks to the several benefits that they offer in the design of new buildings and the management of the existing ones. Frequently, however, BIMs are not available for already built constructions, but, at the same time, the range camera technology provides nowadays a cheap, intuitive and effective tool for automatically collecting the 3D geometry of indoor environments. It is thus essential to find new strategies, able to perform the first step of the scan to BIM process, by extracting the geometrical information contained in the 3D models that are so easily collected through the range cameras.
In this work, a new algorithm to extract planimetries from the 3D models of rooms acquired by means of a range camera is therefore presented. The algorithm was tested on two rooms, characterized by different shapes and dimensions, whose 3D models were captured with the Occipital Structure Sensor[TM]. The preliminary results are promising: the developed algorithm is able to model effectively the 2D shape of the investigated rooms, with an accuracy level comprised in the range of 5 - 10 cm. It can be potentially used by non-expert users in the first step of the BIM generation, when the building geometry is reconstructed, for collecting crowdsourced indoor information in the frame of BIMs Volunteered Geographic Information (VGI) generation.

1. INTRODUCTION

During the last decades, the Building Information Model (BIM) has received increasing attention within the construction sector, due to the several benefits that this method can offer in the design, planning, and construction of new buildings as well as in the management and maintenance of the existing ones. Frequently, however, BIMs are not available for older constructions, and, in some cases, even blueprints are missing. Consequently, there is a strong need to efficiently create BIMs for the already existing and used buildings (Xiong et al., 2013).

At the same time, though, the application of BIM to the retrofitting of existing buildings has to deal with several challenges, caused by the multi-disciplinary nature of information exchange and the wide number of technological components that are needed to obtain an optimal result (Volk et al., 2014), (Donath and Thurow, 2007). In particular, the surveying phase plays an essential role in the process, since an accurate representation of the building geometry is at the basis of a correct generation of the BIM. Therefore, cheap, intuitive and effective tools for automatically collecting the 3D geometry of indoor environments are more and more necessary. In this background, the use of range camera technology can offer significant advantages over traditional methods, thanks to their easiness and quickness of use, especially when not qualified users are involved or when a rapid intervention is required.

Range cameras are indeed active imaging sensors, low-cost and easy-to use, able to natively measure the distances (depths) of several points at high frame rate (30 - 60 Hz), and thus they can be considered, to all intents and purposes, veritable 3D scanners. Furthermore, Simultaneous Localization And Mapping (SLAM) algorithms allow to merge the depth maps captured from different points of view into an overall 3D model automatically and in real-time, by exploiting the depth data and the high frame rate of the range cameras.

It is thus essential to find new strategies, able to extract the information contained in the 3D models so easily collected by the range cameras. This work is precisely included in this background: the aim is to develop a new algorithm to automatically compute the planimetries of environments starting from the 3D models acquired by the range cameras. Indeed, currently the *scan to BIM* process, namely the process of converting 3D model data (point cloud + meshes + texture) into a BIM, is primarily a manual operation, at least in some of its parts, labor-intensive and error-prone (Xiong et al., 2013).

The developed methodology was tested with the Occipital Structure Sensor[TM], the first range camera specifically designed for mobile devices. In particular, the Room Capture application, provided within the Structure SDK by Occipital itself, was used to capture in real-time the 3D models of two indoor environments (two rooms located at the Geodesy and Geomatics Division of the University of Rome "La Sapienza"), characterized by different shapes and dimensions (the maximum lenght of the sides is contained within 8-10 metres) and the corresponding planimetries were extracted by means of the developed algorithm, implemented through some Python scripts and the use of Cloud Compare software (Girardeau-Montaut, 2017).

This paper is thus organized as follows: Section 2. illustrates the procedure used to acquire the 3D models of the rooms, paying specific attention to the capabilities it offers; Section 3. includes the detailed description of the algorithm and Section 4. presents and discusses the obtained results. Finally, in Section 5., some conclusions are drawn and future prospects are outlined.

---

*Corresponding author.

2. COLLECTION OF THE ROOM 3D MODELS

The RoomCapture application was used to collect the 3D models of two rooms. Free and easy to use, this application allows to collect 3D models of indoor environments by simply framing their parts with the Structure Sensor. The model, which appears in live-view on top of the tablet screen, is updated in real-time as soon as new parts of the room are captured. In fact, the SLAM algorithm implemented in the software application is able to continually estimate the position of the moving sensor and consequently to track its trajectory and to co-register automatically the acquired depth maps.

However, since the tracking process can fail, causing an incorrect estimation of the sensor motion during the scanning, it was necessary to identify a general procedure able to limit this kind of error. The incremental alignment of successive frames can indeed generate a progressive accumulation of distortions along the scan path which results in a wrong estimation of the sensor trajectory. When, at the end of the scanning, previously acquired areas are re-captured, they result to be placed in a wrong location, and the estimated trajectory does not close. This may lead to a severe deformation of the final 3D model, whose reconstructed 3D geometry does not correspond in this case to the actual shape of the scanned scene (see Figure 1).
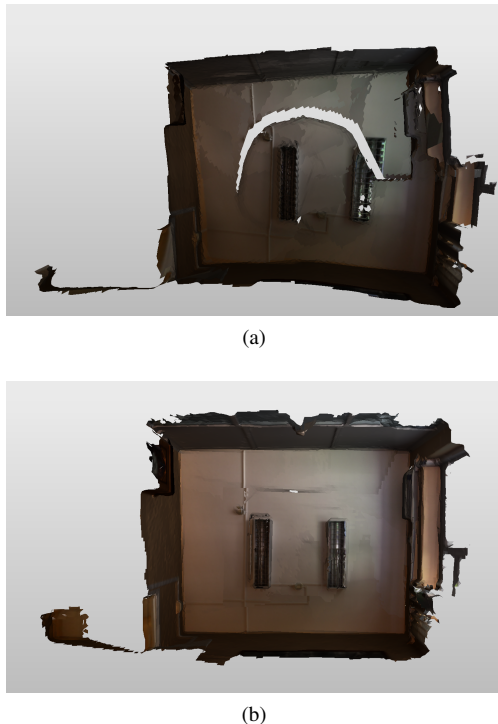

(a)

(b)

Figure 1: 3D model of a room ceiling: (a) not correct tracking, (b) correct tracking.

It is therefore necessary to limit all the unnecessary sensor movements that can bother the tracking process. The 3D models were thus acquired placing the sensor on a swivel pedestal located approximately in the middle of the room, with the sensor maintained as close as possible to the rotation axis. In this way the translation movements were minimized, and the path traveled by the device was mostly circular, as Occipital itself recommends. Finally the dimensions of the scanning volume were set in such a way as to allow the maximum coverage of the upper part of the walls with a minimum upward tilt of the sensor, in order to model exclusively the ceilings of the rooms, namely the parts free of any furniture and obstacles usually located at the floor level. Before starting the scans, the Structure Sensor was calibrated us-

ing the Calibrator application provided by Occipital itself, in order to accurately overlap the 3D data with the corresponding texture.

3. THE ALGORITHM

The developed algorithm extracts a planimetry from the 3D model of a room.

The algorithm takes as input the 3D model of the upper part of a room (see Figure 2); in particular it leverages the information contained in the point cloud, mesh and normal data. This information is stored in the .obj file produced by the RoomCapture application at the end of the scanning process.

The output consists of the 2D coordinates of the vertexes of the perimeter of the polygon describing the room planimetry, together with the equations of the straight lines which model each side. To achieve this result, the following steps are necessary.
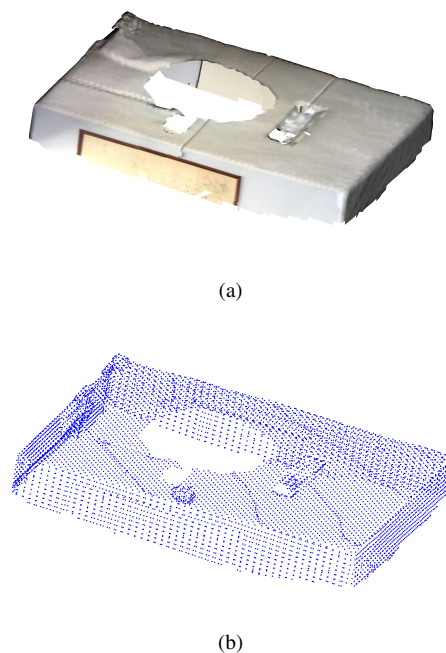

(a)

(b)

Figure 2: Example of the 3D model of a room ceiling acquired with the Structure Sensor: (a) 3D model with texture, (b) 3D point cloud of the room, ready for the RANSAC plane fitting.

3.1 Ceiling plane fitting

First of all, under the hypothesis of rooms with simple geometry (flat ceilings), the plane corresponding to the ceiling is identified through the RANdom SAmple Consensus (RANSAC) algorithm (Fischler and Bolles, 1981); in this way, it is possible to work in a 2D space, by considering exclusively the points of the 3D model that effectively belong to the ceiling (and not to the walls).

In particular, starting with a threshold value of 10 cm (reasonable value), RANSAC is then executed recursively using a threshold of $(10 - n)$ cm, where n is the current number of the iterations (see the example illustrated in Table 1), under those conditions:

- the number of inliers at step n divided by the number of inliers at step $n - 1$ must be > 0.8 (reasonable value to guarantee algorithm stability);

- the angle formed by the plane found at the step $n - 1$ with the plane found at the step n must be lower than a limit angle α (Figure 3). This condition ensures that the recursive

reduction of the threshold does not force RANSAC to fit a plane corresponding to a wall;

- the RANSAC threshold (the threshold distance from the plane to consider a point as inlier or outlier) must be > 2 cm (reasonable value). In several cases, each portion of the 3D model can present a different inclination with respect to the ideal ceiling plane. This circumstance, added to a non-homogeneous distribution of the points, can affect the efficiency of the previous conditions, forcing RANSAC to return a plane that corresponds only to a portion of the ceiling.

Figure 3: The worst-case scenario of the first two steps determinates the maximum angle formed by the planes of two consequents steps.

If the plane of step n violates one of the previous conditions, the algorithm considers the plane computed at the former step as the one corresponding to the ceiling, as shown in Table 1. At this point, the final inlier ensemble must be extended to include all the points effectively belonging to the ceiling, because it typically presents few holes with no points inside (Figure 4) and this behaviour can heavily influence the correct functioning of the whole algorithm later.

The extended inlier ensemble (Figure 5) is thus computed considering all the points which have a distance < 15 cm (reasonable value, selected by the user considering the possible presence of obstacles in the upper portion of the walls, like an air-conditioner) from the identified ceiling plane.

Iteration	Threshold [cm]	Inliers	Outliers	Condition 1	Condition 2
0	10	5395	4581	n.a.	n.a.
1	9	5453	4523	1.01	0.36°
2	8	4833	5143	0.89	0.16°
3	7	5183	4793	1.07	0.14°
4	6	4808	5168	0.93	0.16°
5	5	4713	5263	0.98	0.06°
6	4	4488	5488	0.95	0.04°
7	3	4189	5787	0.93	0.11°
8	2 (!)	3469	6507	0.82	0.03°

Table 1: Iterations of the RANSAC ceiling plane fitting step for the 3D model of a room.
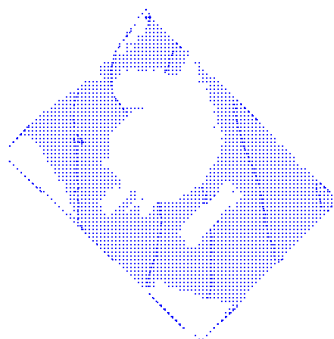
Figure 4: Raw ensemble of inliers, orthogonal projection on the ceiling plane.

3.2 From 3D to 2D

After the removal of the outliers, a new cartesian coordinate system is adopted: two perpendicular straight lines belonging to the
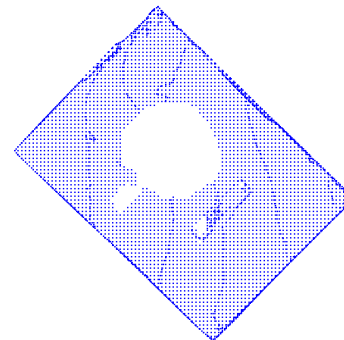
Figure 5: Extended inliers ensemble, orthogonal projection on the ceiling plane.

ceiling plane are selected as the new x and y axes, considering that the scalar product between the new z axis unit vector and the most part of the normal vectors of the inliers (the information about the normals is stored in the 3D model data) must be $>= 0$, otherwise the final planimetry will result a mirror of the true room planimetry.

The original coordinates of the points are thus consequently changed, and the new z coordinate is deleted: all points are orthogonally projected on the ceiling plane, which is by the way the xy plane. At this point a simplification of the new 2D point cloud is necessary in order to remove all the duplicated points generated by the orthogonal projection. An analysis on the operating hardware and the precision level requested should provide a minimum distance that two points must have, to not be considered duplicates (suitably close to zero).

3.3 New triangulation

During the previous step, a new 2D point cloud representing the ceiling is generated and therefore it is necessary to compute new triangular meshes for the new set of 2D points. In particular, the well-known Delaunay Triangulation algorithm is used.

However, in case of rooms with concave geometry, Delaunay Triangulation can erroneously connect points that in the original 3D model were not connected. To avoid this problem, a max edge limit ($m.e.l$) is set: exclusively the triangles with sides $< m.e.l$ will be considered for generating the new meshes (Figure 8). Specifically, the max edge limit value is calculated by means of a statistical analysis performed on the original 3D model: the lengths of all the segments that compose the original triangular meshes are calculated (approximated in centimetres) and the maximum mode (there is a pluri-modal distribution, as shown in Figure 6) with an addition of 1.0 cm (or more, depending on the approximations explained at the end of Section 3.2) is chosen as the value of the maximum edge limit.
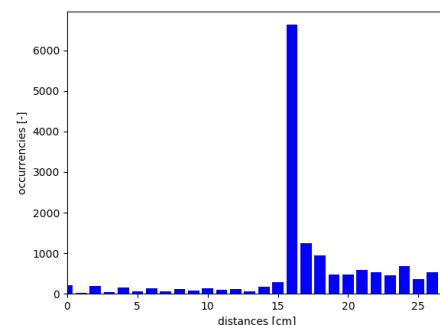
Figure 6: Histogram of distances resulting from the statistical analysis.
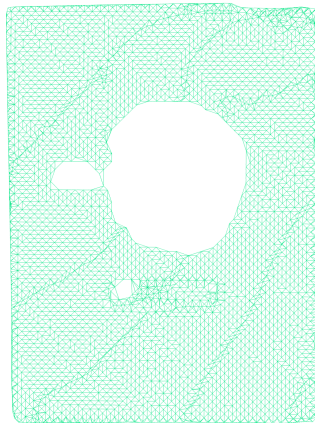
Figure 7: Delaunay triangulation of the 2D point cloud.

Triangles			Segment	Occurrencies	Segment	Occurrencies
0	1	8	**0-1**	**1**	0-8	2
1	8	9	1-8	2	1-9	2
1	2	9	**8-9**	**1**	**1-2**	**1**
2	9	10	2-9	2	2-10	2
2	3	10	**9-10**	**1**	**2-3**	**1**
3	10	11	3-10	2	3-11	2
3	11	12	**10-11**	**1**	3-12	2
3	4	12	**11-12**	**1**	**3-4**	**1**
4	12	13	4-12	2	4-13	2
4	5	13	**12-13**	**1**	**4-5**	**1**
5	6	13	5-13	2	5-6	2
0	5	6	**6-13**	**1**	**0-5**	**1**
0	6	7	0-6	2	0-7	2
0	7	8	0-8	2	**6-7**	**1**
			7-8	**1**		

Table 2: The segments of the 2D point cloud illustrated in Figure 9: the points belonging to the border of the polygon are those which appear only once.

3.4 Segmentation of the perimeter of the Delaunay triangulation

In the Delaunay triangulation, the segment that connects two specific vertexes of the point cloud can belong exclusively to one or two triangular meshes. In the former case, the segment is part of the perimeter. In this way, the points belonging to the perimeter can be thus easily identified by counting the occurences of each segment in all the triangular meshes, as shown in the basic example of Figure 9 and Table 2.
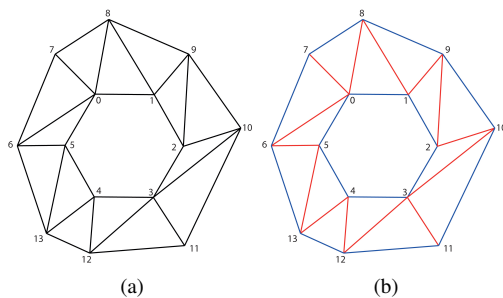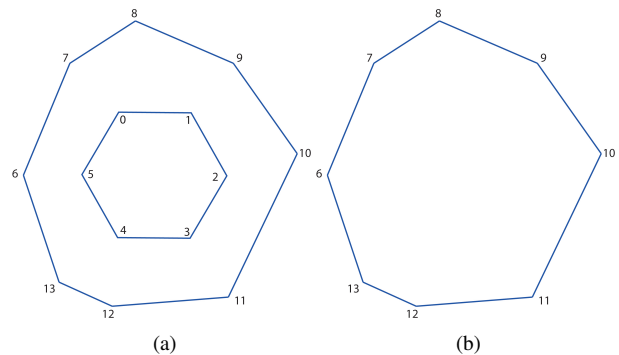

Figure 8: The green filled area is formed by the triangles that belong to the final triangulation, whereas the wireframe belongs to the raw triangulation, its triangular meshes were deleted because of a side longer than the maximum edge limit.

(a)   (b)

Figure 10: Removal of the internal perimeter.

(a)   (b)

Figure 9: Example of a 2D point cloud with points connected by Delauney triangular meshes.

Anyway, by using this method, the border of the ceiling area excluded by the scanning process is selected along with the external perimeter, the only one which interests this work. Thus, to remove this internal perimeter, the found border segments are composed in close polygonal lines; the polygonal line that has the same bounding box of the original 2D point cloud is the one representing the desired external perimeter (Figure 10). Once the segments of the external perimeter are detected, a new 2D point cloud made by all the vertexes belonging to these segments is generated Figure 11.
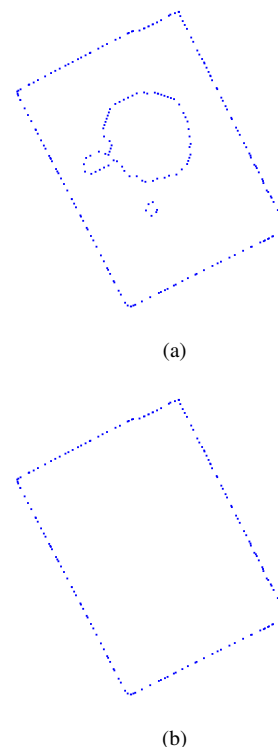

(a)

(b)

Figure 11: Point cloud of the previous example model border, with and without the internal perimeter.

3.5 Planimetry extraction

The planimetries are extracted by identifying the vertexes of the perimeter of the previously computed 2D point cloud. Thus, the RANSAC algorithm is recursively applied to the perimeter points, this time using a linear model to mathematically represent the sides: for each side of the perimeter, the outliers of the previous step are the input data for the next step and so on (Figure 15). In particular, the number n of sides must be specified by the user at the beginning (a threshold of 5 cm is used for all the n iterations). After n steps, if there are no outliers left, RANSAC returns a set of n segments, each one belonging to the found straight line that represents the corresponding side: every segment has the same bounding box of its own inlier ensemble, thus the segments are not connected themselves. Otherwise, if RANSAC cannot return a set of segments without leaving few final outliers, after 100 (arbitrary value, a bigger value can influence the time of execution of the whole algorithm) failed executions it is allowed to leave one outlier, 2 after 200 and so on.

Anyway, in case of non-convex geometry, there is the concrete possibility of considering wrong points among the inliers of a side (see Figure 12): these false inliers do not belong effectively to the side that is being fitted by RANSAC, but they simply intercept the imaginary extension of it. The cause of the false inliers is the width of the segment bounding box and it will create serious problems when the segments are finally turned into a close polygon. To avoid this issue, an additional condition must be imposed: by ordering an inlier ensemble in function of the distances between a generic point and one extreme point, no pair of consecutive points can have a distance higher than the maximum distance measured in a mesh (value computed with the same statistical algorithm of the max edge limit, see Section 3.3).

Finally, the obtained segments are extended on their own straight lines to generate a close polygonal that is the desired planimetry of the room (see Figure 16).
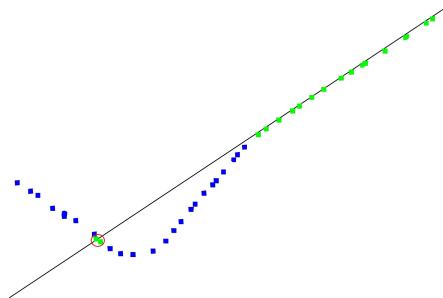
Figure 13: Final result for the convex room: (a) reference planimetry, (b) extracted planimetry.

Figure 12: The false inliers are those highleted with the red circle.

4. RESULTS

As mentioned before, the developed algorithm was tested on the 3D models of two different rooms captured with the Structure Sensor by means of the RoomCapture application.

The obtained planimetries are reported in Figure 13 and Figure 14, where they are visually compared with those obtained through a traditional survey. The results are very promising, even for the concave room (Figure 14), showing a very good agreement between the reference data and the reconstructed planimetries, with an accuracy level comprised in the range of 5 - 10 cm: the developed algorithm is thus able to model effectively the 2D shape of the investigated rooms.
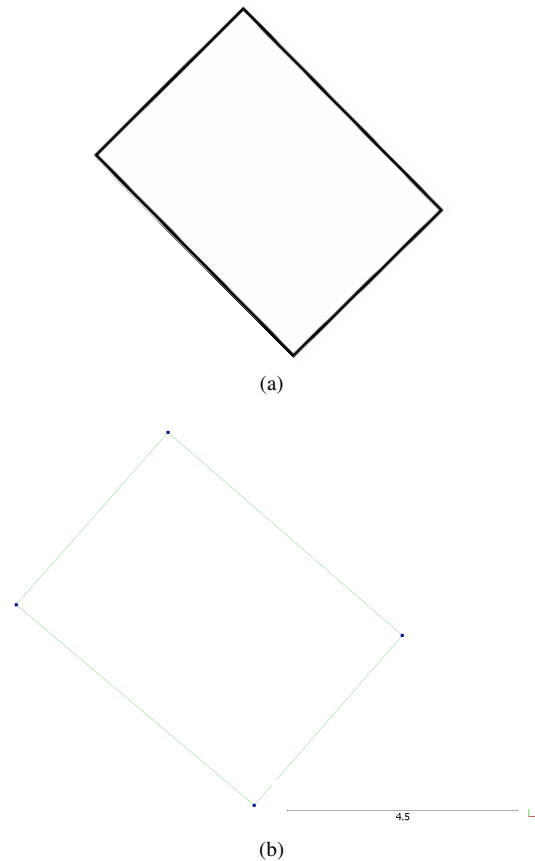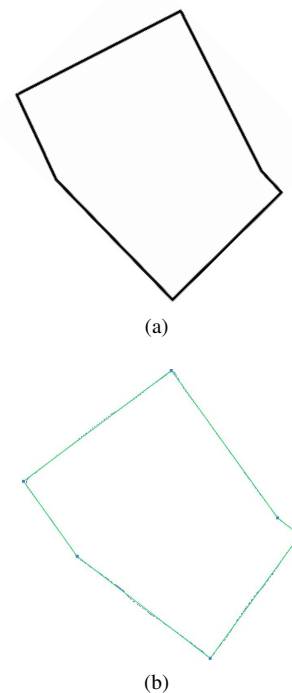
Figure 14: Final result for the concave room: (a) reference planimetry, (b) extracted planimetry.
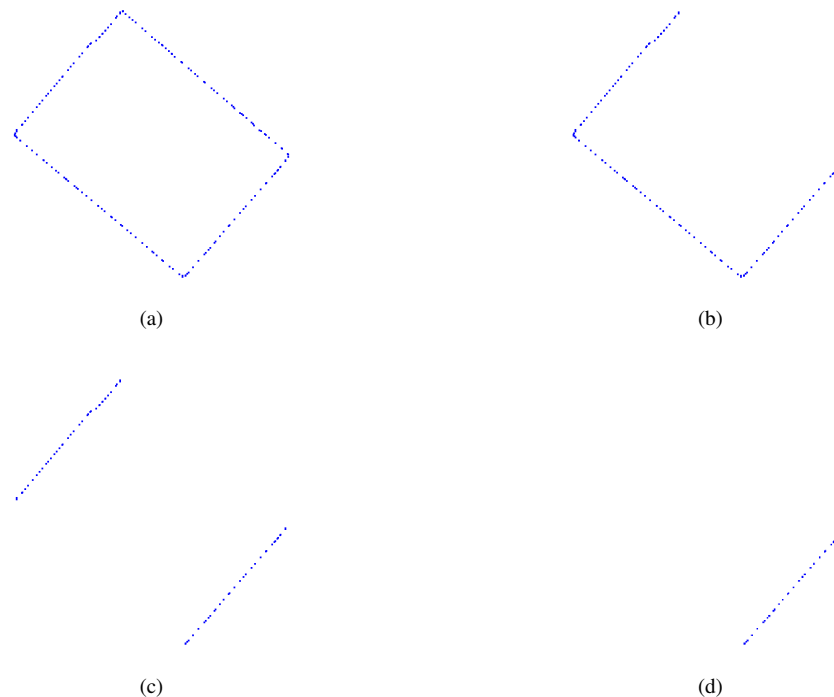
Figure 15: The RANSAC input in the four iterations needed to identify the four sides of the considered room.

5. CONCLUSIONS AND FUTURE WORK

In this work, a new algorithm to extract planimetries from the 3D models of rooms scanned by means of a range camera was developed and implemented. The algorithm was tested on two rooms, characterized by different shapes and dimensions.

The preliminary results are encouraging: the 2D shape of the investigated rooms is effectively reconstructed by the algorithm.

Range cameras and the developed algorithm can be thus potentially used by non-experts for collecting crowdsourced indoor information in the frame of the generation of BIM Volunteered Geographic Information (VGI), at least for the first step of the building geometry reconstruction.

However it is still necessary to deepen the analysis, by testing the algorithm on more rooms and performing a quantitative accuracy assessment.

Finally, it is important to highlight that the planimetry extraction, for now limited to environments with plane ceilings, is the first step towards the complete 3D geometry reconstruction which, as a first approximation, can be easily obtained by capturing also the room height with the range camera.

REFERENCES

Donath, D. and Thurow, T., 2007. Integrated architectural surveying and planning: methods and tools for recording and adjusting building survey data. Automation in Construction 16(1), pp. 19–27.

Fischler, M. A. and Bolles, R. C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24(6), pp. 381–395.

Girardeau-Montaut, D., 2017. Cloud Compare - 3D Point Cloud and Mesh Processing Software - Version 2.8.1. Open Source Project.

Volk, R., Stengel, J. and Schultmann, F., 2014. Building Information Modeling (BIM) for existing buildings Literature review and future needs. Automation in construction 38, pp. 109–127.

Xiong, X., Adan, A., Akinci, B. and Huber, D., 2013. Automatic creation of semantically rich 3D building models from laser scanner data. Automation in Construction 31, pp. 325–337.
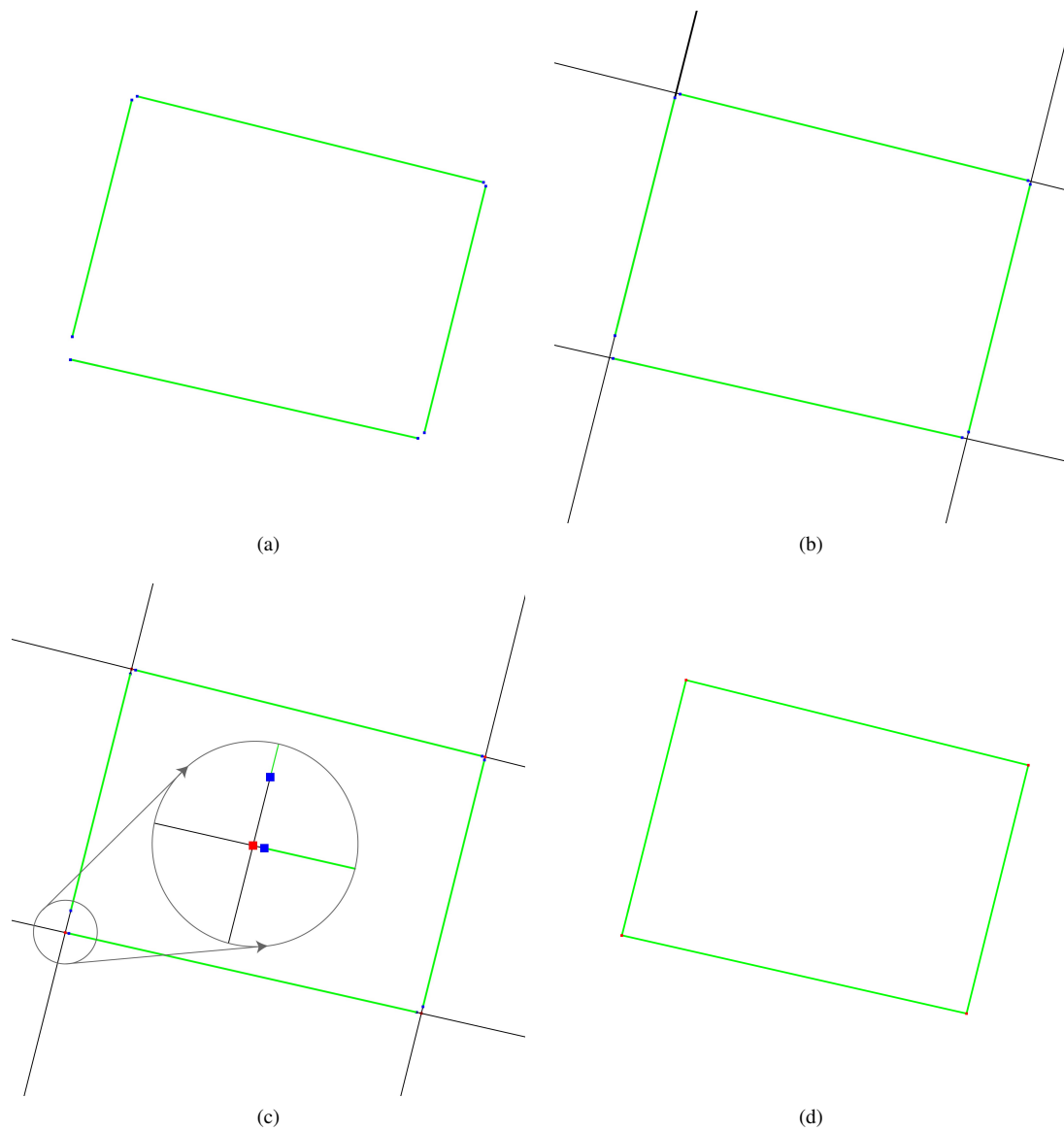
(a)

(b)

(c)

(d)

Figure 16: The steps necessary to extract the planimetry from the side segments identified through the RANSAC linear fitting.