

Omnidirectional image capture on mobile devices for fast automatic generation of 2.5D indoor maps

Giovanni Pintore CRS4, Pula (CA), Italy giovanni.pintore@crs4.it www.crs4.it/vic	Valeria Garro ISTI-CNR, Pisa, Italy valeria.garro@isti.cnr.it vcg.isti.cnr.it	Fabio Ganovelli ISTI-CNR, Pisa, Italy fabio.ganovelli@isti.cnr.it vcg.isti.cnr.it
Marco Agus CRS4, Pula (CA), Italy marco.agus@crs4.it www.crs4.it/vic	Enrico Gobbetti CRS4, Pula (CA), Italy enrico.gobbetti@crs4.it www.crs4.it/vic	

Abstract

We introduce a light-weight automatic method to quickly capture and recover 2.5D multi-room indoor environments scaled to real-world metric dimensions. To minimize the user effort required, we capture and analyze a single omnidirectional image per room using widely available mobile devices. Through a simple tracking of the user movements between rooms, we iterate the process to map and reconstruct entire floor plans. In order to infer 3D clues with a minimal processing and without relying on the presence of texture or detail, we define a specialized spatial transform based on catadioptric theory to highlight the room's structure in a virtual projection. From this information, we define a parametric model of each room to formalize our problem as a global optimization solved by Levenberg-Marquardt iterations. The effectiveness of the method is demonstrated on several challenging real-world multi-room indoor scenes.

1. Introduction

The problem of determining the architectural structure and a simplified visual representation of indoor environments has attracted a lot of attention in recent years, with proposed solutions ranging from mostly manual floor plans sketchers (e.g., [29]) to automatic methods that process high-density scans (e.g., [21]). Devices such as laser scanners often represent the most effective but expensive solution for a dense and accurate acquisition [35]. Therefore, their use is often restricted to specific application domains, such as Cultural Heritage or engineering, and they are hardly applicable to time-critical applications. The emergence of Kinect-style depth cameras has lowered the cost

of methods based on active sensors, producing impressive results even for building-scale reconstruction [33]. Moreover, 3D reconstruction methods based on multiple images have become quite popular [1, 20] and, in certain situations, the accuracy of dense image-based methods is comparable to laser sensor systems at a fraction of the cost [28]. However, they typically require non-negligible acquisition and processing time, and most of these approaches often fail to reconstruct surfaces with poor texture detail. In summary, the common issue with all these classes of methods is that they require considerable effort to produce simplified structured models of buildings from the high-density data. Commodity mobile devices, such as phones and tablets, nowadays enable any user to perform fast multimodal digital acquisition [12] and effective information extraction [6]. The creation of simplified indoor models using reduced human effort has a variety of applications, ranging from free-viewpoint navigation using high-quality texture-mapped models [3] to the management of building evacuations or real-time security systems [13].

In this paper, we introduce a light-weight method to quickly capture and recover 2.5D multi-room indoor environments scaled to real-world metric dimensions (see Fig. 1). The central idea in this work is to minimize both user and computational effort by capturing and analyzing a single omnidirectional image per room using the built-in capabilities of modern mobile devices.

Approach. For many typical indoor environments exhibiting a piecewise-planar structure, a single equirectangular image can contain enough information to recover the room shape. By exploiting theories commonly employed in catadioptric systems [2], we define a geometric transform for virtually projecting the room in order to highlight its struc-

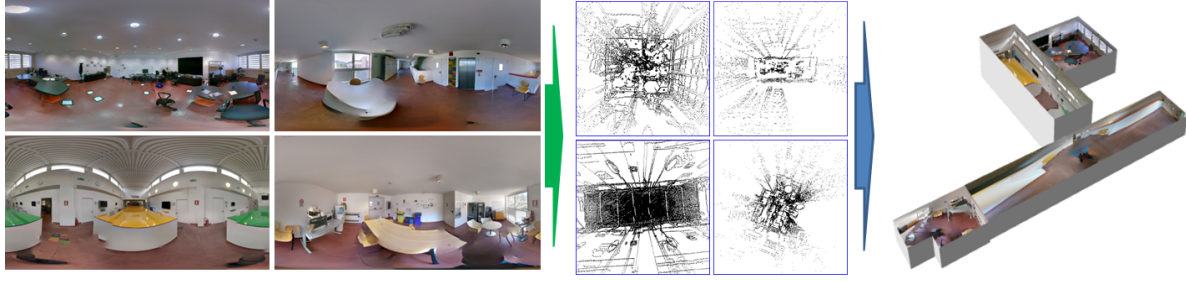


Figure 1: We take as input one omnidirectional image for each room. To infer 3D clues without externally calculated 3D points or MVS data, we introduce a transform to project the image gradient map to a 2D plane. As a result, we obtain a 3D representation of the captured environment along with its visual representation through spheremaps.

tural features. From this information, we create a parametric model of the room to which we apply a global optimization to yield the room’s shape. Having the value of the height of the observer, we obtain the shape of the room and its height in real-world dimensions. Furthermore, if the mobile device is equipped with IMU (Inertial Measurement Unit), by tracking the user’s movements between rooms we can iterate the method to map and reconstruct the entire floor plan.

Main contributions. Our approach automatically builds multi-room models from omnidirectional images, even when the walls in the scene do not form right angles. We introduce a spatial transform for equirectangular images bringing the problem into a 2D space and recovering a prior parametric model of the rooms from these images. We propose a voting scheme to estimate wall height and to identify a set of boundary points in the image, exploiting these to enable the solution of the reconstruction problem as a global optimization. Since our approach is not computationally demanding, it is possible to fully implement the entire acquisition and reconstruction pipeline on a mobile device.

Advantages. Our method allows mobile device users to quickly measure and sketch an indoor environment. A single panoramic image per room can be easily obtained by off-the-shelf guided applications – a much simpler approach than multi-view methods. Instead of relying on costly off-line processing, our proposed approach provides immediate processing with an automatic and light-weight floor map reconstruction method. The proposed approach returns accurate results even for scenes with surfaces lacking in texture and details, unlike MVS (Multi View Stereo) methods. The whole pipeline returns rooms in real-world units, allowing the composition of multi-room models without manual interventions. In contrast to many of the previous approaches (see Sec. 2), neither strong *Manhattan World* constraints, nor further 3D information (e.g., original unstitched images, externally calculated 3D points, MVS data) are needed. Finally, our method for panorama analysis can be applied to enhance structure classification in other applications [3, 15]. For instance, as indoor panoramas are

themselves gaining increased popularity (e.g., Google Maps tours), developing geometry extraction methods bridges the gap from purely visual navigators to 3D reconstruction.

Limitations. Although our method is not limited to a Manhattan World, it does assume that the room is piecewise planar. Also, since the method requires omnidirectional images, whenever the generation of such images fails the method cannot be applied. Moreover, relying on a single viewpoint per room it simplifies capture, but makes the method sensitive to strong occlusions. Despite these limitations, the method is very effective in a variety of indoor environments, ranging from private houses to large public spaces, as demonstrated by our results (see Sec. 8).

2. Related Work

Our approach combines and extends state-of-the-art results in many areas of computer vision and mobile capture. Here, we discuss the methods that are most related to our technique.

Floor plan extraction. Previous works in floor plan extraction can be classified in different categories according to the level of user input required (automatic, or semi-automatic), to the geometric constraints (Manhattan World assumption or other structural regularities), and according to the input data. User-assisted approaches have long proven effective for floor plan reconstruction [26, 18, 24], but they have the disadvantages of requiring significant and repetitive user inputs and being prone to errors from device mishandling and manual editing. To overcome these limitations, over the past few years a number of fully automated approaches have been presented based on simplifying geometric assumptions and/or employing additional 3D information; many of these also assume a prior knowledge of the scene. Regarding the geometric assumptions, a number of methods exploit structural regularities such as planarity or orthogonality as priors [16], like the Manhattan World assumption [19, 18], which states that all the surfaces are aligned with three dominant directions, typically corresponding to the X, Y, and Z axes. With respect to the input data, many effective methods model 2D planar maps of indoor structure starting from 3D

point clouds. The first such systems were derived for processing indoor laser scan data, employing bottom-up region growing [14], Hough line detection [31], the RANSAC algorithm [27], and plane fitting [25]. Alternative techniques take advantage of RGB-D cameras that allow a live capture of both depth and appearance information at affordable cost, but their distance acquisition is rather limited in range and resolution. A common strategy is based on consecutive frames alignment [17] by jointly optimizing over matching depth and color information. This approach leads to sequential error propagation that can be managed by loop-closure algorithms. A global alignment of frames [22, 32] can provide more robust acquisitions. Furukawa et al. [3] reconstruct the 3D structures of moderately cluttered interiors by fusing multiple depth maps (created from images) through the solution of volumetric Markov Random Fields under the heavily constraining Manhattan World assumption. However regularization in MRF is only based on pairwise interaction terms, and thus susceptible to noisy input data. Cabral et al. [3] extend the work of Furukawa et al. [10] by extracting complementary depth cues to stereo from the single images. All aforementioned methods obtain 2D floor plans from 3D data originating from different sources. Our technique is different because the input it requires is only a single equirectangular image for each room to be reconstructed, and it automatically computes a precise 2D floor plan with real-world metric dimensions by using as prior information only the height at which the spherical map is acquired.

Analysis of panoramic images. The rapid growth of omnidirectional image photography applications, such as *Android Photo Sphere* developed by Google, has led to extensive utilization of automatically stitched omnidirectional images in a variety of circumstances – both for displaying outdoor scenes and indoor rooms. With respect to scene understanding, omnidirectional images have been successfully exploited for localizing objects [30], calibrating catadioptric systems [2], recognizing view points [34], and recovering indoor structures [23]. Although most of the studies dealing with the omnidirectional images are focused on catadioptric view, many useful properties can be extended to equirectangular images [11]. Our method exploits these theories to describe a visual model of the scene based on the spherical projection and minimizing geometric constraints. Furthermore, a other few methods [5, 36] have recently been proposed for reconstructing indoor floor plans from omnidirectional images, but these techniques, unlike ours, require additional user input and they are limited by the Manhattan World assumption.

3. Approach Overview

Similarly to other effective approaches [3, 9], for each room image we perform a first classification to identify the

ceiling and the floor. Since not all omnidirectional images are well stitched, and because of the particular challenges posed by imagery from real-world indoor environments (poor lighting, ambiguity in vanish points recognition), an accurate classification of the image is hard to obtain without integrating externally calculated 3D points and prior knowledge of the orientation of the walls. To address this problem we use the theory for central panoramic systems [11] to define a spatial transform G_h (Sec. 4) which, under specific conditions, returns 3D Cartesian points from angular coordinates in the spheremap. Applying the transform for an unknown wall height through a specialized voting scheme we calculate a set of points S_m with a high likelihood of being on real room boundaries, coupled with an estimation of the wall height.

We apply the transform to the image gradient map, projecting its values to a plane, and arrange the projected points in a 2D array. This 2D array is a sort of *fingerprint* of the shape (e.g. Fig. 1 center) where points that are on the walls edges tend to concentrate their projection in the same place, as well as points not satisfying the hypothesis of the transform G_h do not have a real 3D correspondence and are sparsely distributed.

By the analysis of this 2D array we obtain a prior model of the room resulting in a parametric representation which varies in a constrained angular space $S(\theta, \gamma)$ (Fig. 4 left). Hence we formalize our problem as a global optimization on the measures S_m , resulting in the final shape of the room in real-world metric units. Since the method is fully automatic and assumes the use of a mobile device (although it is applicable for single omnidirectional images coming from different sources), we can extend it to reconstruct entire floor plans by adding minimal information regarding the user’s direction of movement (Sec. 7).

4. Transform definition

We take as input an *equirectangular* image of the room – i.e., a spherical image which has 360 degrees longitude and 180 degrees latitude field of view. We assume that the input image is already aligned to the gravity vector and that each corner of the room is visible. These conditions are usually satisfied by spheremaps generated with the aid of sensor fusion in modern mobile devices (e.g., *Google Camera with Photo Sphere*, *Autostitch* [4]), and they are commonly adopted by systems like *Google Street View*. Since we also assume that the acquisition is done with a mobile device the height of the observer’s eye is known (easy to estimate with a quick calibration step); the device can also provide a simple tracking of the user’s movement between rooms.

To classify the floor and the ceiling in the image we start with an approach similar to [3]. A super-pixels based segmentation method [7] is combined with a geometric reasoning classification [9], exploiting the texture homogeneity

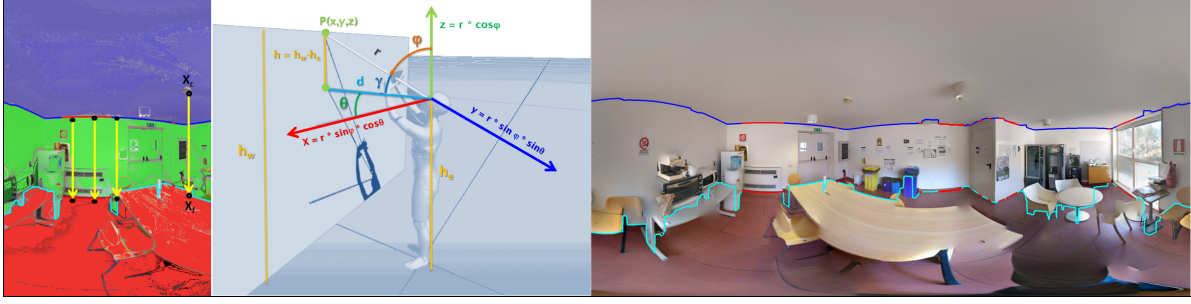


Figure 2: Left: mapping transferring the points between the ceiling and the floor (real case simplified for the exposition). Center: each point on the (*spheremap*) image can be mapped into a 3D space through the transform 5. From each point (θ, γ) in the image we can generate a 3D point when its height h is known. Right: boundary points extracted during the initial classification step. The points marked in red are *strong* correspondences.

prevalent in indoor scenes and labeling the top and bottom parts of the image as ceiling and floor (blue and red zones respectively in Fig. 2 left). According to this classification model the floor is related to the ceiling through a planar homology $H_{c \rightarrow f}$ (Fig. 2 left), which can be recovered given the location on the image of any pair of corresponding ceiling/floor points (\bar{x}_c, \bar{x}_f) [8]. This approach is very effective when features are lines but less reliable in many real-world case of indoor spherical omnidirectional images. Therefore, Cabral et al. in [3] enforce the label assignment by introducing 3D/MVS information, which is calculated externally from the original sparse image set, and by introducing a priori knowledge of the height of the observed walls. From this first classification (ceiling, walls, floor) we obtain two sets of pixels $I(\bar{x}_c)$ and $I(\bar{x}_f)$ (for the ceiling and for the floor) that have a high probability of containing the floor-wall and ceiling-wall intersections, respectively.

As in [8], we do not have a priori a pair of points (\bar{x}_c, \bar{x}_f) . Instead of trying to infer one from additional 3D information or imposing the Manhattan World assumption, we introduce a specialized *Transform* G_h and room model to solve our problem.

The origin of this room's model is the position of the ideal observer, where the abscissa and ordinate of the image respectively represent the azimuth θ and the tilt γ of the view's direction. For the rest of this explanation we assume that the *mapping between angles and pixels is implicit*, since this transformation in a equirectangular image is supposed to be linear. Each point in the (*spheremap*) image can be mapped in a 3D space through the following spherical coordinates (see Fig. 2 center)

$$G(r, \theta, \varphi) = \begin{cases} x = r * \sin \varphi * \cos \theta \\ y = r * \sin \varphi * \sin \theta \\ z = r * \cos \varphi \end{cases} \quad (1)$$

We can appropriately convert with respect to the direction viewing (Fig. 2 center) through the following relations

$$\begin{aligned} \sin \varphi &= \cos \gamma \\ \cos \varphi &= \sin \gamma \\ r &= d / \cos \gamma \end{aligned} \quad (2)$$

If we introduce the assumption that the height z is a constant value h for all points the distance d of the observer to the wall is

$$d = \frac{h}{\tan \gamma} \quad (3)$$

and we also have:

$$z = h = r * \sin \gamma \Rightarrow r = h / \sin \gamma \quad (4)$$

and substituting for r in Equation 1 we obtain the function:

$$G_h(\theta, \gamma) = \begin{cases} x = h / \tan \gamma * \cos \theta \\ y = h / \tan \gamma * \sin \theta \\ z = h \end{cases} \quad (5)$$

The function G_h maps all the points of the equirectangular image in 3D space as if their height was h . We will use G_h with one of the values:

$$h = \begin{cases} -h_e & \text{floor} \\ h_w - h_e & \text{ceiling} \end{cases} \quad (6)$$

where h_e is the height of the center of the omnidirectional image (the eye of the observer) and h_w the height of the wall. If we knew the wall height h all the pixels in $I(\bar{x}_c)$ and $I(\bar{x}_f)$ would be mapped to their actual 3D position. This observation leads us to a test for assessing the likelihood that a given value h is indeed the actual wall height.

For each image column j , we apply the function G_h to the pixels belonging to $I(\bar{x}_c)$ and $I(\bar{x}_f)$ (that is, $I(\bar{x}_c)|_j$ and $I(\bar{x}_f)|_j$). If h is the actual wall height, then the XY coordinates of the points on the edges on the wall (both on the floor and on the ceiling) should be the same, since the wall is assumed to be vertical. Unfortunately, the initial classification also returns many pixels in other positions, like the furniture edges – the cyan pixels in Fig. 2 right. However,

we rely on the fact that most likely $I(\bar{x}_c)$ and $I(\bar{x}_f)$ do contain points on the wall.

For each pair of pixels $(c_j, f_j) \in I(\bar{x}_c)|_j \times I(\bar{x}_f)|_j$ we define:

$$d_h(c_j, f_j) = \text{dist}_{XY}(G_h(c_j), G_h(f_j)) \quad (7)$$

where dist_{XY} is the Euclidean distance on the XY plane. Note that $d_h(c_j, f_j)$ is small in two cases: either because h is near the actual value and both the pixels on the wall (or in the unlikely case that the edge detector returned false positives on the floor and the ceiling at the same XY position), or h is not near the actual value and the pair is a false positive. Therefore, we consider the most likely h the one that maximizes the following term:

$$d(h) = \sum_{\forall j} \text{count}\{(c_j, f_j) \mid d_h(c_j, f_j) < \tau\} \quad (8)$$

where τ is a metric threshold that we set to 5 cm in our experiments.

The optimization could be performed in several ways, for instance with RANSAC or even by gradient descent. However, since we reduced our problem to a single variable h , the search space can reasonably be limited to between 2 m and 10 m. Thus, we can afford to perform a voting scheme iterating h over the interval in 2 mm steps, which is below the tolerances for indoor construction, so we avoid even slim chances of running into a local minimum. When h is found we select the subset of couples $(\hat{x}_c(\theta, \gamma), \hat{x}_f(\theta, \gamma))$ for which $d_h(c_j, f_j) < \tau$ and mark them as strong correspondences (in red in Fig. 2 right). These pairs identify a set of image points $S_m(\theta, \gamma)$ that with an high likelihood belong to the room boundaries. We will exploit them in final reconstruction step, in conjunction with the room parametric model described below.

5. Parametric model

Most of the studies dealing with spherical panoramic images are focused on the catadioptric view [2], but many the same theorems can be applied to all omnidirectional images as well. In a spherical panoramic image, a line in the world is projected onto the unit sphere as an arc segment on a great circle [11].

Starting from these assumptions we apply the *Transform* $G_h(\theta, \gamma)$ in Eq. 5 to the *Canny* edge map, projecting points from polar coordinates $\in S(\theta, \gamma)$ to \mathbb{R}^2 through a projective plane π_{xy} .

Projected points form an array $\Pi(x, y)$ (see Fig. 3 left), whose parametric space is quantized in metric dimensions (i.e., centimeters). Although not all values have a real 3D correspondence, the points having a high likelihood of being on the real room's boundaries tend to *accumulate* their

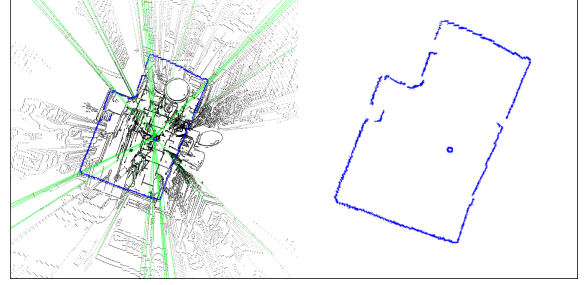


Figure 3: Left: Simplified illustration of the transform defined by Eq. 5. Projected data contains both noise, a sheaf of 2D lines (green) with center in the origin of the room and a *fingerprint* of the room shape (blue lines). Right: Detail (scaled and enhanced for printing) of the accumulation peaks.

projection in the 2D array $\Pi(x, y)$. Furthermore, mapping the problem to a 2D Cartesian space greatly simplifies the detection of shapes, as geometric lines (conics in image space) become lines in the projective plane π_{xy} .

Since $\Pi(x, y)$ can also be considered as a 2D image, we can easily highlight a basic model of the room shape with the Hough transform. Indeed, as illustrated in Fig. 3 left (green lines), vertical edges in the 3D scene tend to become a sheaf of lines Γ in \mathbb{R}^2 with center in the origin of the room, whereas the ceiling and floor boundaries accumulate their projection in same or adjacent positions, describing the set of segments Λ in \mathbb{R}^2 . Once we have removed sparse points from the image of $\Pi(x, y)$, we choose the intersections of segments Λ that intersect or have a small distance from a line $\in \Gamma$, since we expect many of these radial lines to intersect the shape corners. As a result we obtain a subset of segments $\Lambda_{int} \subset \Lambda$ in \mathbb{R}^2 whose intersections $\{p_1, \dots, p_n\}$ with $p_i \in \mathbb{R}^2$ correspond to the n corners of a reasonable model of the room shape (see Fig. 3 right).

From the intersections $\{p_1, \dots, p_n\}$ we estimate their approximate positions in polar coordinates $\in S(d, \theta)$ (see Fig. 4 left). Since d depends on γ and h according to Eq. 3, once we choose one of the two boundary planes $z = h$ with its related h from Eq. 6, each boundary (ceiling or floor) of the room can be represented in equirectangular coordinates as a set of corners $\{c_1(\theta_1, \gamma_1), \dots, c_n(\theta_n, \gamma_n)\}$ with $c_i(\theta_i, \gamma_i) \in S(\theta, \gamma)$ (see Fig. 4 top-left).

6. Room shape extraction

To obtain the reconstruction of the real room layout we fit the parametric model described in Sec. 5 to the measurements $S_m(\theta, \gamma)$ (see Sec. 4). Given the m measurements $S_m(\theta, \gamma) = \{\hat{x}_{s_1}, \dots, \hat{x}_{s_m}\}$ we generate their corresponding $T_m(\theta, \gamma)$ values related to the room parametric model. As previously described in Sec. 4, the set $S_m(\theta, \gamma)$ is composed of pairs of points $(\hat{x}_{c_j}(\theta, \gamma), \hat{x}_{f_j}(\theta, \gamma))$ (related respectively to positions in the ceiling and the floor) sharing the same θ_j value. For each point in $S_m(\theta, \gamma)$ we generate

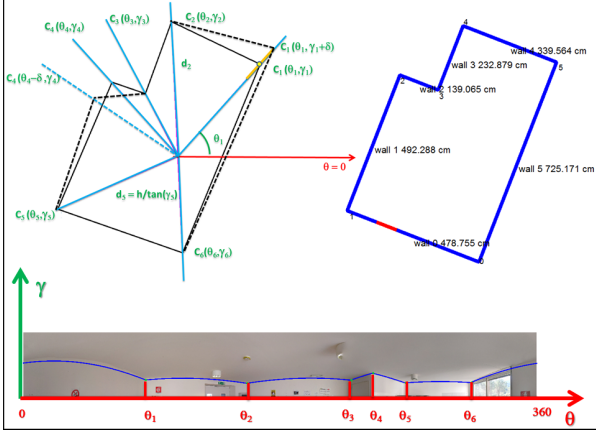


Figure 4: Left: we generate all possible shapes from a set of angles varying in an opportune range (e.g., $\pm\delta$). From the model values in angular space (bottom) we sample the corresponding T_m samples to be compared with the S_m measurements. Right: final reconstruction of the room in metric units.

a point in our parametric model, acquiring its corresponding distance value d through ray casting and converting its 3D coordinates to angular coordinated through the inverse of Eq. 5.

We perform a global optimization of the $T_m(\theta, \gamma) = \{\hat{x}_{t_1}, \dots, \hat{x}_{t_m}\}$ samples generated by varying the $2n$ parameters of the model to estimate the set of parameters $R(\theta_1, \gamma_1, \dots, \theta_n, \gamma_n)$ that best describe the real shape of the room. The problem can be formalized as a non-linear least squares problem (Eq. 9), solvable with a Levenberg-Marquardt algorithm (LMA).

$$R(\theta_1, \gamma_1, \dots, \theta_n, \gamma_n) = \underset{j=1}{\operatorname{argmin}} \sum^m \|\hat{x}_{s_j} - \hat{x}_{t_j}\|^2 \quad (9)$$

Mathematically it is not uncommon to find the parameters wandering around near the minimum in a flat valley of complicated topology, since the minimum is at best only a statistical estimate of $R(\theta_1, \gamma_1, \dots, \theta_n, \gamma_n)$.

In our case, since all parameters are represented by angles and the initial values are strictly bounded to a closed polygon and a short angular range, a very limited number of iterations are always sufficient to ensure convergence to a good solution. (Fig. 4 right).

7. Floor Plan Generation

The method described in the previous Section can be iterated to map and reconstruct a multi-room structure with the addition of minimal tracking of the user’s movements through the mobile device’s IMU (see Sec. 8 for details). We track the approximative direction of the user with respect to the Magnetic North when he/she moves from a room to another; we also spatially reference each

spheremap during the acquisition (i.e., the direction of image center is known w.r.t. the Magnetic North). Once we have roughly identified the exit and entrance doors in the omnidirectional images displayed in the GUI, we then identify doors in the images with conventional CV methods (vline/rect detection), without the need to track/infer the whole user’s path (see Fig. 5). In order to obtain compact floor-plans and a better alignment between walls, we check for close corners between adjacent rooms (Fig. 5 yellow dots) and we slightly tune the door positions to minimize the distance between these corners. The interconnections between matching doors are stored in a graph of the scene, then for each matching door between two adjacent rooms r_j to room r_{j+1} we calculate the 2D affine transform $M_{j,j+1}$ representing the transform from the coordinates of room r_{j+1} to room r_j .

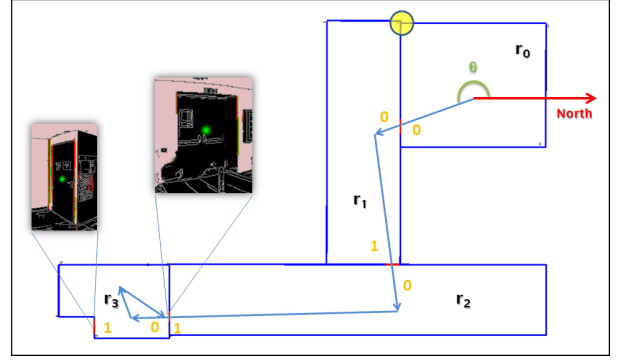


Figure 5: We align each new room to an initial r_0 , calculating the path to reach the starting room as a set of transforms representing the passages encountered while moving from the aligned room to r_0 .

We chose a room r_0 as the origin of the floor plan. Then, for each aligned room r_j we calculate: the path to the origin room as a set of transforms representing the passages required to reach it; and the whole transformation to the origin room coordinates (Fig. 5). Since each room is already scaled to the same metric coordinates, the final floor plan result is automatically aligned and scaled as well, without any manual intervention.

8. Results

Data acquisition. To demonstrate the effectiveness and accuracy of our method, we implemented a minimal Android application (compatible with version 4.4 and higher) to capture a multi-room indoor scene. This application keeps track of the user’s movements between rooms and acquires the sphere-map of each environment. In addition, it estimates the height of the ideal eye (see model Fig. 2 right) with respect to the floor through a simple calibration at a known distance. Although different solutions are available to capture the spherical omnidirectional images, we choose

Scene Name	Features		Area error		Wall length error		Wall height error		Corner angle error		Editing time MagicPlan
	Area [m^2]	Np	MP	Ours	MP	Ours	MP	Ours	MP	Ours	
Office H1	720	10	2.95%	1.78%	35 cm	15 cm	2.0 cm	1.2 cm	0.8 deg	0.8 deg	26m32s
Building B2	875	25	2.50%	1.54%	30 cm	7 cm	6.0 cm	1.5 cm	1.5 deg	1.5 deg	42m18s
Commercial	220	6	2.30%	1.82%	25 cm	8 cm	12.0 cm	2.7 cm	1.5 deg	1.0 deg	28m05s
Palace	183	3	16.86%	0.20%	94 cm	5 cm	45.0 cm	1.3 cm	1.8 deg	0.5 deg	15m08s
House 1	55	5	21.48%	2.10%	120 cm	16 cm	15.0 cm	4.7 cm	13.7 deg	1.2 deg	25m48s
House 2	64	7	28.05%	1.67%	85 cm	8 cm	18.0 cm	3.5 cm	15.0 deg	0.5 deg	32m25s
House 3	170	8	25.10%	2.06%	115 cm	15 cm	20.0 cm	4.0 cm	18.0 deg	1.5 deg	29m12s

Table 1: Comparison vs. ground truth and other methods. We indicate the floor area and the number Np of input panorama images/rooms. We show the comparison between our method and MagicPlan (MP) in terms of area error, wall length and wall height maximum error encountered. Finally, we show the additional editing time needed by MagicPlan to achieve a result comparable to ground truth.

to use the Google Camera and its related *Photo Sphere* module to make the results easily replicable. Through this application we save the floor plan as a scene graph of interconnected rooms, storing for each room the following components: an equirectangular image covering a view of 360×180 degrees of the room; the direction with respect to the Magnetic North of the image center; the direction in the sphere map of the door to the previous room; and the direction of the door to the next room. By comparing the direction of the doors the application automatically calculates and stores the interconnection between rooms and the path between them. We tested our technique on a variety of single rooms acquired with the Android-based system just described and also with data from more general sources, especially to facilitate comparison to other approaches.

Implementation. The method is implemented on Android and is based on freely available tools. The first segmentation and classification step (Sec. 4) is implemented with *OpenCV*¹, as in prior work [7, 3]. *OpenCV* is also used for all the standard operations on 2D images (using C++ and Android calls).



Figure 6: Apartment with 7 rooms (Tab. 1 House2). On the left, the blueprint assumed as ground-truth with its real measures inserted by the designer. On the right, our reconstruction.

Evaluation. Table 1 summarizes the results obtained for indoor structures whose real dimensions are known; the structures were acquired with the mobile Android application described above. We also show results with publicly available

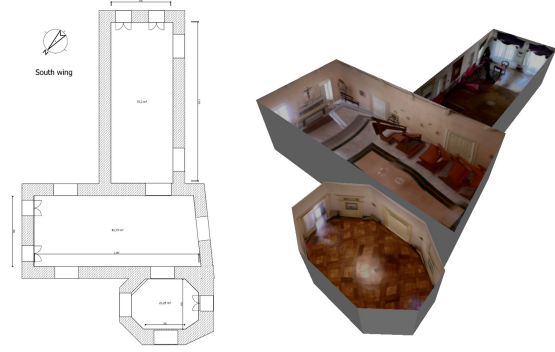


Figure 7: South wing of an ancient palace (reference removed for blind review - Tab. 1 Palace). On the left the floor plan assumed as ground-truth with its real measures manually acquired. On the right our reconstruction.

omnidirectional images that were already studied with other single-image methods alternative to ours, to compare the results. Since the goal of the method is the metric reconstruction rather than obtaining high accuracy in texture-mapping, the typical Pixel Classification Error (percentage of pixels that disagree with ground-truth label) is not a suitable measurement to evaluate the accuracy of the predictions, nor for a direct comparison with state-of-the-art methods [10] employing 3D/MVS data. We instead choose to demonstrate the accuracy of our method by adopting as ground-truth the real world dimensions of the acquired indoor structures and comparing according to metric units. In Tab. 1 we compare the ground truth, our method, and the latest version of MagicPlan, which integrates some of the features proposed in [26, 24]. We have a non-negligible increase in performance in Manhattan World environments, with similar results for wall lengths, heights and angles. Moreover, our method shows significant improvements when acquiring more general environments (e.g., area errors of 0.2-2.1% vs. 16.9%-28.0%; similar improvements for linear measures and angles). In addition, MagicPlan (and Yang et al. [36]) require extra editing steps, taking from a few seconds to over 30 min of additional user time. In Fig. 6 we show the reconstruction of a complete multi-room envi-

¹<http://www.opencv.org>

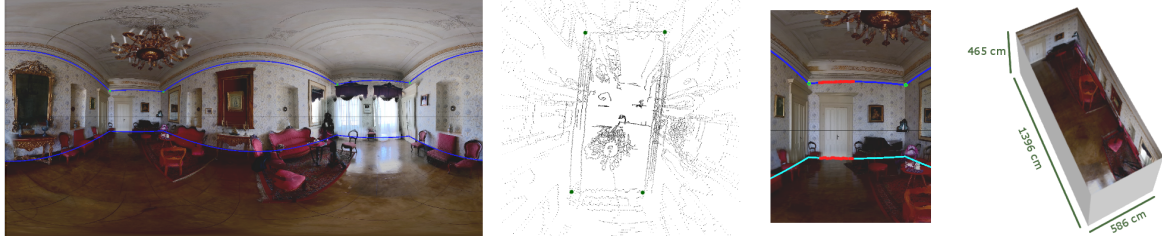


Figure 8: Room from the Palace dataset. The green points show some of the *strong* point pairs used in the analysis.

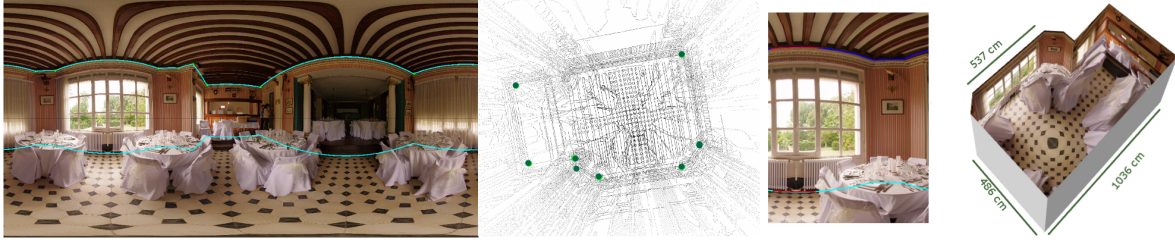


Figure 9: Chateau de Sermaise, France, courtesy of Flickr. Omnidirectional image used for comparison with other methods. The result was calculated automatically by our method in *5.5 seconds*. Yang et al. [36] obtain a comparable result on the same dataset by manual modeling in *71 seconds*.

ronment (House 2 of Tab. 1), with several Non-Manhattan World walls. Assuming as ground-truth the blueprint, slight differences in the layout are due to the presence of balconies and a different furnishing of the bathroom and the kitchen compared to the initial project. In Fig. 7 our method successfully performs the reconstruction of a Non-Manhattan world environments, such as the private chapel and the octagonal state room.

Figure 8 shows a detail from the Palace dataset acquired with our mobile system. Unlike the other test cases we used, this one exhibits smoothed ceiling edges making it difficult to identify the real boundaries from the image. Nevertheless, the method correctly recognizes as ceiling boundary the upper extremities of the vertical walls, returning an accurate metric reconstruction (the estimated height of the walls is 460 cm) at the cost of a less accurate texture mapping. In Fig. 9 we compare our method with [36]. Our system returns a metric reconstruction of the environment automatically in about *5 seconds*, in contrast to a comparable result obtained by Yang et al. [36] in *71 seconds* through manual modeling. Although no data is available from mobile sensors in this case, we estimated a reasonable height of the ceiling of about 5 meters by assuming an average camera height of 165 cm. Since not all corners are visible in the image, our system recovers a fitting model with 8 corners (green dots), still finding the best closed polygon which represents the shape and avoiding this type of failure case. A second portion of the scene environment with different ceiling height is also visible in the right part of the image and is correctly classified by the system as a different room. On the other hand, we can see that our method is unusable in presence of curved walls or if the ceiling is supported by

arches, as showed by the failure case illustrated in Fig. 10.



Figure 10: Failure case: room with the ceiling supported by arches. Although the walls boundaries looks like conics in the spheremap, as they are like projections of lines, the transform reveals their geometry, resulting in a failure of the model detection.

9. Conclusions

We presented a light-weight method to rapidly map and reconstruct many typical indoor environments. Our design exploits the features of modern mobile devices, including motion and location sensors and the ability to generate panorama images. Since the approach is not constrained by a Manhattan World assumption and the prior model is defined at run-time, the method can be extended to sloped ceilings – for example, with a refined implementation of the voting scheme. A straightforward improvement would be using multiple omnidirectional images for each room, to cover cases where the entire perimeter is not visible from a single point. This could be done, for example, by combining our method with real-time approaches for fisheye image matching [15].

Acknowledgments. This work has received funding from the European Union’s Seventh Framework Programme for research, technological development and demonstration under grant agreement no 607737 (VASCO). We also acknowledge the contribution of Sardinian Regional Authorities

References

- [1] Autodesk. 123D Catch. www.123dapp.com/catch. 1
- [2] J. Bermudez-Cameo, L. Puig, and J. Guerrero. Hypercatadioptric line images for 3D orientation and image rectification. *Robotics and Autonomous Systems*, 60(6):755 – 768, 2012. 1, 3, 5
- [3] R. Cabral and Y. Furukawa. Piecewise planar and compact floorplan reconstruction from images. In *Proc. CVPR*, pages 628–635, 2014. 1, 2, 3, 4, 7
- [4] Cloudburstresearch. Autostitch for Android, 2015. www.cloudburstresearch.com/. 3
- [5] T. K. Dang, M. Worring, and T. D. Bui. A semi-interactive panorama based 3D reconstruction framework for indoor scenes. *Comput. Vis. Image Underst.*, 115(11):1516–1524, 2011. 3
- [6] K. Dev and M. Lau. Democratizing digital content creation using mobile devices with inbuilt sensors. *IEEE CG&A*, 35(1):84–94, 2015. 1
- [7] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *Int. J. Comput. Vision*, 59(2):167–181, 2004. 3, 7
- [8] A. Flint, C. Mei, D. Murray, and I. Reid. A dynamic programming approach to reconstructing building interiors. In *Proc. ECCV*, pages 394–407. Springer, 2010. 4
- [9] A. Flint, D. Murray, and I. Reid. Manhattan scene understanding using monocular, stereo, and 3d features. In *Proc. ICCV*, pages 2228–2235, 2011. 3
- [10] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski. Reconstructing building interiors from images. In *Proc. ICCV*, 2009. 3, 7
- [11] C. Geyer and K. Daniilidis. A unifying theory for central panoramic systems and practical implications. In *Proc. ECCV*, pages 445–461, 2000. 3, 5
- [12] Google. Tango, 2014. www.google.com/atap/projecttango/. 1
- [13] J. Guest, T. Eaglin, K. Subramanian, and W. Ribarsky. Interactive analysis and visualization of situationally aware building evacuations. *Information Visualization*, 2014. 1
- [14] D. Hähnel, W. Burgard, and S. Thrun. Learning compact 3D models of indoor and outdoor environments with a mobile robot. *Robotics and Autonomous Systems*, 44(1):15–27, 2003. 3
- [15] C. Hane, L. Heng, G. H. Lee, A. Sizov, and M. Pollefeys. Real-time direct dense matching on fisheye images using plane-sweeping stereo. In *Proc. 3DV*, pages 57–64, 2014. 2, 8
- [16] V. Hedau, D. Hoiem, and D. Forsyth. Recovering the spatial layout of cluttered rooms. In *Proc. ICCV*, pages 1849–1856, 2009. 2
- [17] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox. RGB-D mapping: Using depth cameras for dense 3D modeling of indoor environments. In *Proc. ISER*, pages 477–491, 2010. 3
- [18] Y. M. Kim, J. Dolson, M. Sokolsky, V. Koltun, and S. Thrun. Interactive acquisition of residential floor plans. In *Proc. ICRA*, pages 3055–3062, 2012. 2
- [19] D. C. Lee, M. Hebert, and T. Kanade. Geometric reasoning for single image structure recovery. In *Proc. CVPR*, pages 2136–2143, 2009. 2
- [20] Microsoft. Photosynth. photosynth.net/. 1
- [21] C. Mura, O. Mattausch, A. Jaspe Villanueva, E. Gobbetti, and R. Pajarola. Automatic room detection and reconstruction in cluttered indoor environments with complex room layouts. *Computers & Graphics*, 44:20–32, 2014. 1
- [22] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. W. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Proc. ISMAR*, pages 127–136, 2011. 3
- [23] N. Ozisik, G. Lopez-Nicolas, and J. J. Guerrero. Scene structure recovery from a single omnidirectional image. In *Proc. ICCV Workshops*, pages 359–366, 2011. 3
- [24] G. Pintore and E. Gobbetti. Effective mobile mapping of multi-room indoor structures. *The Visual Computer*, 30, 2014. 2, 7
- [25] V. Sanchez and A. Zakhor. Planar 3D modeling of building interiors from point cloud data. In *Proc. ICIP*, pages 1777–1780, 2012. 3
- [26] A. Sankar and S. Seitz. Capturing indoor scenes with smartphones. In *Proc. UIST*, pages 403–412, New York, NY, USA, 2012. ACM. 2, 7
- [27] R. Schnabel, R. Wahl, and R. Klein. Efficient RANSAC for point-cloud shape detection. In *Computer Graphics Forum*, volume 26, pages 214–226, 2007. 3
- [28] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proc. CVPR*, volume 1, pages 519–528, 2006. 1
- [29] Sensopia. Magicplan, 2014. www.sensopia.com. 1
- [30] P. Sturm. A method for 3d reconstruction of piecewise planar objects from single panoramic images. In *Proc. OMNIVIS*, pages 119–119, 2000. 3
- [31] F. Tarsha-Kurdi, T. Landes, and P. Grussenmeyer. Hough-transform and extended ransac algorithms for automatic detection of 3D building roof planes from lidar data. In *ISPRS Workshop on Laser Scanning and SilviLaser*, volume 36, pages 407–412, 2007. 3
- [32] D. Thomas and A. Sugimoto. A flexible scene representation for 3D reconstruction using an RGB-D camera. In *Proc. ICCV*, pages 2800–2807, 2013. 3
- [33] T. Whelan, M. Kaess, M. Fallon, H. Johannsson, J. Leonard, and J. McDonald. Kintinuous: Spatially extended kinectfusion. In *Proc. RSS Workshop on RGB-D*, 2012. 1
- [34] J. Xiao, K. A. Ehinger, A. Oliva, and A. Torralba. Recognizing scene viewpoint using panoramic place representation. In *Proc. CVPR*, pages 2695–2702, 2012. 3
- [35] X. Xiong, A. Adan, B. Akinci, and D. Huber. Automatic creation of semantically rich 3D building models from laser scanner data. *Automation in Construction*, 31(0):325 – 337, 2013. 1
- [36] H. Yang and H. Zhang. Modeling room structure from indoor panorama. In *Proc. VRCAI*, pages 47–55, 2014. 3, 7, 8