Mobile Mapping and Visualization of Indoor Structures to Simplify Scene Understanding and Location Awareness

Giovanni Pintore[1], Fabio Ganovelli[2], Enrico Gobbetti[1], Roberto Scopigno[2]

[1]Visual Computing, CRS4, Italy
[2]Visual Computing Group, ISTI-CNR, Italy
{giovanni.pintore,gobbetti}@crs4.it {ganovelli,scopigno}@isti.cnr.it

Abstract. We present a technology to capture, reconstruct and explore multi-room indoor structures, starting from panorama images generated with the aid of commodity mobile devices. Our approach is motivated by the need for fast and effective systems to simplify indoor data acquisition, as required in many real-world cases where mapping the structure is more important than capturing 3D details, such as the design of smart houses or in the security domain. We combine and extend state-of-the-art results to obtain indoor models scaled to their real-world metric dimension, making them available for online exploration. Moreover, since our target is to assist end-users not necessarily skilled in virtual reality and 3D objects interaction, we introduce a client-server image-based navigation system, exploiting this simplified indoor structure to support a low-degree-of-freedom user interface. We tested our approach in several indoor environments and carried out a preliminary user study to assess the usability of the system by people without a specific technical background.

Keywords: Mobile Systems; Scene Understanding; Scene Reconstruction; Smart Environments; Safety and Security

1 Introduction

Indoor environments are where humans spend most of their time, and the current evolution towards a digitally assisted society puts them at the center of thriving research and development efforts, aimed at provide a large variety of assistive technologies for improving quality of life. The most direct realization of this trend is the appearance of indoor location based services (Indoor LBS), such as indoor maps or indoor routing [14]. Moreover, recent years have witnessed an increasing interest in *smart homes*, that is, the integration of technology and services through home networking for a better quality of living. The possibility of automatizing the management of a house by introducing sensors and actuators handled by AI is now a reality and gave rise to a consistent body of literature and a flourishing industry. Smart homes are not considered just a costly technological gadget anymore, but as a way to make people who are impaired in some way,

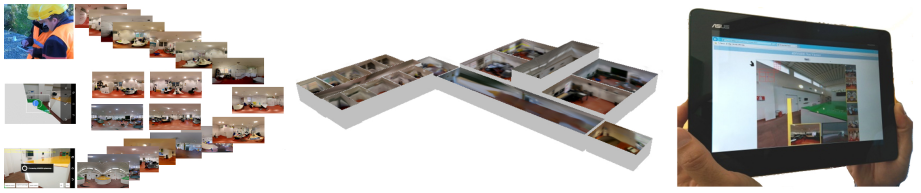

Fig. 1. Our method at a glance: from a set of panoramic photos (left) our system computes a textured 3D model of the indoor boundaries (center) which can be explored with a low-degree-of-freedom web application (right).

or exposed to some risk, to be self-sufficient in their home. It is needless to add that, in our aging society, the audience is becoming broader and broader. One obstacle to diffusion of smart homes is the cost to develop them [28]. For this reason several simulators have been proposed to design and test a smart home before putting the hardware into place [1, 16, 20, 21, 18]. Such an acquire, simulation, and test approach is used in a variety of other applications, and in particular in the security domain for visually defining and assess concepts and measures for protecting buildings in case of dangerous events [15].

One of the major limiting factors in the creation of such simulators, and in creating a large variety of digitally-assisted applications targeting indoor environments, is the lack of suitable 3D mock-ups of the target environment. Since detailed CAD models often do not exist for many buildings, and/or the as-built situation is often very different from the recorded plans, quick and fast ways to create structural and visual information of indoor environments are paramount. Moreover, CAD models alone would not suffice for a variety of needs, such as location awareness, which requires models that are photorealistic enough to recognize real places by just looking at them.

For a widespread use, users should be able to create, access and share digital mock-ups of buildings without requiring the assistance of computer experts to model them. Although devices such as laser scanners often represent the most effective but expensive solution for a dense and accurate acquisition [29], their use is often restricted to specific application domains, such as Cultural Heritage or engineering, as well as this solution requires expensive equipment and very specialized personnel. 3D reconstruction methods based on multiple images have become quite popular and, in certain situations, the accuracy of dense image-based methods is comparable to laser sensor systems at a fraction of the cost [26]. However, they typically require non-negligible acquisition and processing time, and most of these approaches often fail to reconstruct surfaces with poor texture detail. Moreover, the common issue with all these classes of methods is that they require considerable effort to produce structured models of buildings from the high-density data.

Current mobile devices combined with computer vision techniques offer a very attractive platform to overcome these problems. Mobile devices have in-fact become increasingly attractive due to their multi-modal acquisition capabilities and growing processing power, which enables fast digital acquisition, interactive scene understanding, and effective information extraction [8]. Integrating a cap-

ture and explore pipeline in a mobile device would boost the development of next-generation, collaborative natural user and visual interfaces for many critical applications, such as the management of building evacuations or real-time security systems [15].

In this work, we propose a novel hybrid approach that uses a cost-effective capture technique based on stitched panoramic images and sensor data acquired with modern mobile devices. The proposed method is capable of extracting a measurable model of room structures, as well as a traversable visual representation, which can be explored with an image-based visual exploration system.

Approach. We start from the assumption that for many typical indoor environments a single equirectangular image of a room can contain enough information to recover the architectural structure. We combine and extend state-of-the-art works to quickly capture indoor environments by acquiring a single equirectangular image per room and a graph of the scene, using panoramic capture tools and instruments commonly available on mobile devices. On the captured scene we perform an automatic reconstruction based on the application of catadioptric theories [2]. Once the multi-room scene has been reconstructed it can be edited and shared and by multiple users, and interactively explored through a platform-independent *WebGL* viewer.

Main contributions. We introduce an integrated system to simplify indoor building capture, mapping and its photorealistic exploration, without actually involving costly and time consuming 3D acquisition pipeline or manual modeling. We combine automated approaches [4, 22] with an aided user interface to obtain a 3D textured model in real-world metric dimensions even when automatic tasks fail, such as when the typical assumptions of geometric reasoning [17, 10] are not verified (i.e., high piecewise planarity and a large fraction of the room's boundaries unequivocally detectable in the image). Furthermore, since the obtained multi-room environment is already structured and simplified, we exploit an assisted image-based rendering approach to support interactive navigation, instead of using pre-computed video sequences when moving from a room to another one (e.g., [25, 9]), thus reducing lag and network bandwidth.

Advantages. The proposed system is targeted to provide a fast and effective method to capture and share real indoor environments to end-users not necessarily skilled in virtual reality and 3D objects interaction. Such a pipeline automatically returns rooms in real-world units only entering manually the height of the observer's eye h_e (such value remains valid for the whole acquisition); moreover it allows the composition of multi-room 3D models with minimal user interaction. In contrast to many of the previous approaches (see Sec. 2), neither further 3D information (e.g., original unstitched images, externally calculated 3D points, MVS data) nor heavy *Manhattan World* [6] assumptions are needed.

Limitations. Relying on a single image per room makes the method sensitive to strong occlusions. Despite these limitations (although common to almost all related approaches), the method is always effective if at least each corner position is visible in the image either on the ceiling or on the floor room's boundaries.

2 Related work

Reconstruction of indoor scenes. 3D reconstruction of architectural scenes is a challenging problem in both outdoor and indoor environments. Compared to building exteriors, the reconstruction of interiors is complicated by a number of factors. For instance, visibility reasoning is more problematic since a floor plan will in general contain several interconnected rooms. In addition, interiors are often dominated by clutter, surfaces that are barely lit and texture-poor walls. This results in a very challenging reconstruction and in noticeable high frequency rendering artifacts, negatively affecting the quality of visualization. Approaches range from 3D laser scanning [19] to image-based methods [11]. These methods produce high resolution 3D models, which are often an overkill for a large branch of applications, especially those focused on the structure of a building rather than the details of the model.

The use of modern mobile devices has become a promising approach for short-range 3D acquisition and mapping, as witnessed by the well known *Google Project Tango* and *Microsoft Kinect*. However, rooms larger than a few meters, for example a hotel hall, are outside the depth range of these sensors and make the acquisition process more time consuming. Mobile multi-room mapping is useful in many different real-world scenarios, such as smart homes, security management and building protection, etc., mainly to enable non-technical people to create models with enough geometric features for simulations [23] or enough information to support interactive virtual tours [25].

In recent times there has been a renewed interest in omnidirectional images. Applications such as *Android Photo Sphere*, developed by Google, has led to extensive utilization of automatically stitched spherical images in a variety of scenarios. These images are nowadays generated by specific devices, both for entertainment (e.g. *Samsung Gear 360*, *Ricoh Theta S* or general monitoring applications (e.g. fisheye cameras). Cabral et al. [4] extend the work of Furukawa et al. [11] to label indoor structures from omnidirectional images, by exploiting externally calculated depth cues to stereo from the unstitched images. Most of the studies dealing with spherical panoramic images are focused on catadioptric view [2], but many theorems can be applied to all panorama images with practical implications. Following the theories of catadioptric systems [12], Pintore et al. [22] describe a visual model of the scene based on the spherical projection and minimizing geometric constraints. Although their method only acquires the room's footprint and not its entire 3D content, it has the merit of using a single panoramic image per room and no further user interaction, except for handling the capture of multiple rooms. Nevertheless the method is effective only under specific assumptions, such as the presence of large portions of the walls borders easily detectable on the ceiling and the floor.

3D navigation. Navigating a 3D model can be done in several ways. In our case as in many general cases the problem consists in controlling position and orientation of a virtual camera inside a 3D scene, providing the corresponding image. Therefore is broken down to a problem of human machine interaction, as well as a problem of real-time rendering. In classical FPS (first person shooter)

videogames [5] the viewer movement is controlled by keys that map to the four main directions (forward, backward, left and right), generally referred as *wasd* mode. Although this interaction paradigm is very effective for a swift control of the viewing parameters in interaction-critical setting, it becomes too much engaging for more generic uses. This is because it requires both hands, and it is simply not doable with modern touch devices, that have neither keyboard nor mouse.

On the other end of the spectrum there are approaches where the view is very constrained and therefore the interaction is minimal. For example there are approaches based on the use of collection of registered images resulting from a SfM reconstruction [27]. In these approaches the views on the 3D scene are those corresponding to the images. The transition between neighbor images can be represented with different strategies. In [27] are rendered by fading between images, or by using partial 3D point cloud reconstruction as outputted by SfM [24]. [13] and [3] combine the images with a accurate 3D scanning of the location improving them by further processing.

As middle solution between free and constrained navigation are pure image-based techniques that use panoramic images. One well known example is *Google StreetView*, where the viewer can move from on panoramic images to its neighbors by clicking on specific spots. A similar approach is also used in a mobile fashion in [25] for navigating indoor environments. In this case a 360 video panorama for each room is used, in which the user may pan left of right to explore the scene, whereas the transition between panoramas are shown with a video shot while moving from a room to the next. The same combination of panoramas and videos is more generically used in [9] for virtual environments, where both panoramas and videos are obtained by external photorealistic rendering engines. Following the trend of combining mobile and image-based methods we propose the model described in the following sections.

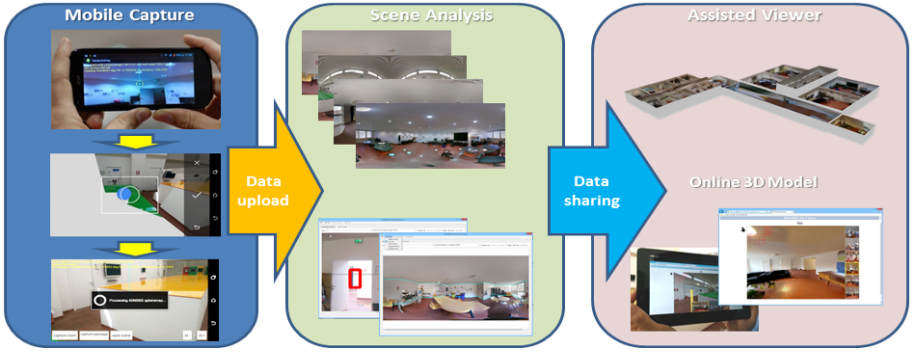

Fig. 2. Mobile Capture module: an interactive mobile application to capture and generate a spherical image for each room, map the multi-room structure by tracking the user movement and recover metric information. **Scene Analysis module:** a scene analysis tool to reconstruct the 3D model and to create a navigable scenegraph. **Assisted Viewer module:** a client/server viewer based on WebGL to navigate inside the 3D model through an assisted navigation interface.

3 System overview

The proposed pipeline can be outlined as three modules (Fig. 2): **Mobile Capture**, **Scene Analysis** and **Viewer**.

- **Mobile Capture** an interactive mobile application exploiting image and sensor data to:
  - generate 360 degrees equirectangular images of each indoor environment
  - map the multi-room structure by tracking the user movements to generate a navigable scenegraph
  - recover the metric information about the structure
  - upload the acquired data to the Scene Analysis module
- **Scene Analysis** A scene analysis tool interfaced with a server to:
  - collect the data from the acquisition modules to reconstruct the floor plan structure in real world metric dimensions
  - create the navigable scenegraph for the Viewer's server
- **Viewer** A platform-independent viewer to browse the data on the server and explore the reconstructed scene:
  - based on WebGL and running on standard browsers
  - supporting interactive exploration through a low-DOF navigation interface, suitable for touch interaction on mobile devices

4 Methods

4.1 Scene capture

Starting from a convenient point of view (from which an observer can see all the room's corners) the user acquires a spheremap of the surrounding environment through the *Mobile Capture* module, generating for each room an *equirectangular* image – i.e., a spherical image which has 360 degrees longitude and 180 degrees latitude field of view (Fig. 3). Equirectangular images are widely used by mobile stitching tools (e.g., *Google Camera with Photo Sphere*), or adopted by systems like Google *Street View*. Moreover these images can be generated by specific hardware, such as compact 360 video cameras (e.g.*Ricoh Theta S*), surveillance circuits, unmanned vehicles instruments. After the acquisition of a room is completed, he/she moves to the next one, tracking the approximative moving direction with respect to the Magnetic North by the aid of the mobile device's IMU. Each spherical image results also spatially referenced, since the equirectangular projection is an angular map and the direction of the image's center is also known through the mobile sensors (w.r.t. the Magnetic North). From these information we easily locate the directions respectively of the exit door from a room and of the entrance door in the next one, and consequently the doors matching between adjacent rooms. At processing time these angular information about doors is thus exploited to recover their local 2D Cartesian coordinates, resulting in a navigable graph of the scene and a spatial mapping of the rooms (see Sec. 4.2).

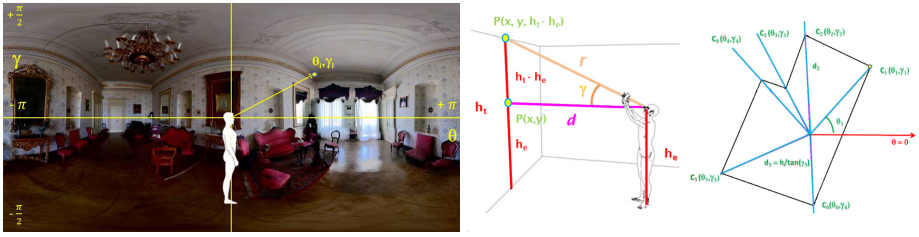

Fig. 3. Left: The equirectangular image represent all possible views for an observer ideally located in the center of the spheremap, identified by the angles θ and γ. Center: h_e is the physical height of the center of the equirectangular image (the ideal eye of the observer) and it is the only value externally entered and not automatically estimated. h_t is the height of the targeted point in respect to the floor plane. Right: The transformation G_h returns real Cartesian coordinates if the height h is known.

4.2 Scene reconstruction

Once the acquired environment has been uploaded to a server we reconstruct the 3D scene starting from the analysis of the single images. Each room is represented by an equirectangular image (see Fig. 3 left), covering every visible point seen by the observer, hence each pixel in the spheremap defines a specific direction of view identified by two angles: θ and γ. The angle θ is the heading of the targeted point respect to the magnetic North, defined as a rotation around the ideal axis between the Sagittal and Coronal planes of the observer. The angle γ (observer tilt) is instead the rotation around the axis between the Coronal and Transverse planes.

Following the basis of the geometric reasoning proposed in [22] we assume that each point in the spheremap can be mapped in 3D space through the following spherical coordinates (see Fig. 3 center):

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

We can appropriately convert with respect to the direction of view through the following relations

$$\begin{aligned} \sin\varphi &= \cos\gamma \\ \cos\varphi &= \sin\gamma \\ r &= h/\sin\gamma \end{aligned} \tag{2}$$

substituting for in Equation 1 we obtain the function:

$$G_h(\theta,\gamma) = \begin{cases} x = h/\tan\gamma * \cos\theta \\ y = h/\tan\gamma * \sin\theta \\ z = h \end{cases} \tag{3}$$

Assuming the walls are vertical G_h maps all the points of the equirectangular image in 3D space as if their height was h:

$$h = \begin{cases} -h_e & floor \\ h_t - h_e & target \end{cases} \tag{4}$$

where h_e is the height of the observer's eye, located in an ideal center of the spheremap, and it is the only value not automatically estimated by our system. Instead h_t is the height of a targeted point (Fig. 3 right) calculated in respect to the floor plane. Since h_e is a constant quantity during the whole acquisition, we obtain the floor plan reconstruction scaled in real-world metric dimensions just entering this value or estimating it with a quick calibration step through the mobile device sensors. Observing Fig. 3-*right* is clear how the shape of the room is known if the angular positions θ_i and γ_i of the *i-corners* (we assume as room's corner position in the image its location on the intersection between wall and ceiling or between wall and floor) are known from the equirectangular map, since the resulting points obtained applying eq. 3 are actually the Cartesian coordinates of the room.

Although semi-automatic approaches exist to identify the room shape (see Sec. 2), they are hardly practicable in many real-world contexts, since they requires externally calculated 3D data or they work only under certain strictly conditions (i.e., heavy piecewise planarity and walls boundaries easily detectable in the image). After performing a preliminary automatic recognition based on [22],

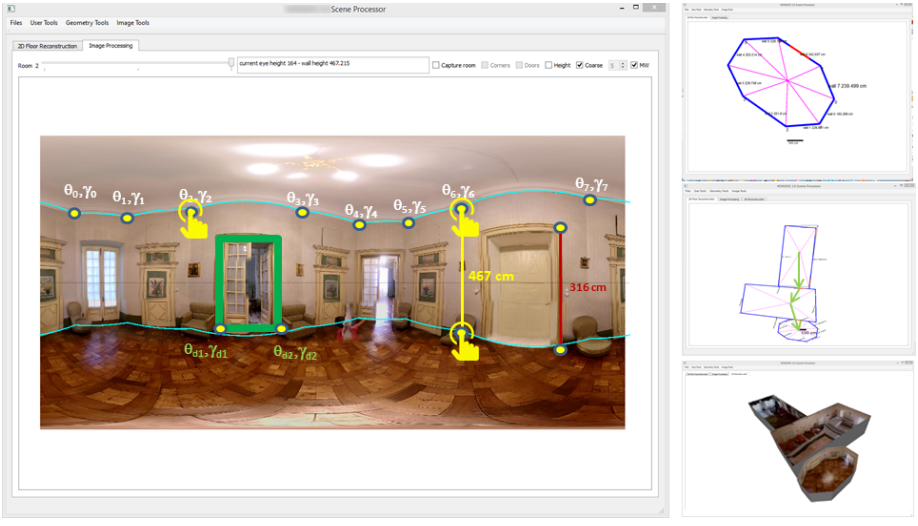

Fig. 4. Server side interface. Left: this tool enables several preliminary scene analysis tasks, like metric measurements on the image (i.e. wall height in yellow, door height in red) and room's corners identification starting from the automatic boundaries detection (i.e. cyan contours). Top right: reconstruction in real-world scale of the analyzed room. Center right: user's path (green) stored during capturing and relative rooms displacement. Bottom right: final 3D textured reconstruction of the acquired floor plan (T2 Tab 1).

we enable the user to integrate the reconstruction by the interface illustrated in Fig. 4. Through this tool we can easily supervise the automatic reconstruction (Fig. 4 cyan borders) and eventually adjust room's corners, acquire measures (Fig. 4 yellow and red segments) and other features of the room architecture,

just establishing a correspondence between picked points (i.e. θ, γ couples) on the screen and geometric points in the indoor scene.

As matter of fact in the spherical panoramic imaging a line in the world is projected onto the unit sphere as an arc segment on a great circle. The arc segment on a great circle forms a curve segment in an omnidirectional image [12]. As practical implication several operation can be performed by the user to integrate the reconstruction or to extract further information. Assuming vertical walls the distance d from the observer to the wall (Fig. 3 center) is a constant value between the ceiling and the floor boundary. We can automatically calculate this quantity d in respect to the floor combining equation 3 and 4, obtaining the room height (i.e. yellow distance on corner 6 in Fig. 4 left):

$$d = \frac{-h_e}{\tan \gamma_f} \tag{5}$$

where γ_f is the view direction of a targeted point on the intersection between floor and wall planes. If the target point is on the walls ceiling boundary we can estimate the height of the room h_w as:

$$h_w = h_t = d * \tan \gamma_c + h_e \tag{6}$$

where γ_c is the view direction of a targeted point on the intersection between wall and ceiling. Moreover the application of eq. 3 to $c_1(\theta_1, \gamma_1), \cdots, c_n(\theta_n, \gamma_n)$ returns the positions $p_1, \cdots, p_n \in \mathbb{R}^3$, defining the room's shape (Fig. 3 right). if the automatic reconstruction fails because one or even all corners are not identified the user can recover the task just indicating with a simple click their position in the image click (i.e. yellow circle Fig. 4 left).

In a similar way the positions of the doors are individuated in the image (Fig. 4 green marker), and this information is integrated with the doors matching derived by the acquired user's path (Fig. 4 right center). According with this matching we define a connection between two rooms,e.g. r_j and r_{j+1}, as a couple of doors that fundamentally are the same door expressed in different coordinates. We calculate a transform $M_{j,j+1}$ between r_{j+1} and r_j just comparing the corresponding door extremities. From the rooms connectivity graph we calculate for each room $r_p \in (fr_1 \dots r_N)$ the path to a global origin room r_0 (usually the first one acquired), as the multiplication of the transforms $\{M_1 \cdots M_p\}$ representing the passages encountered to reach r_0.

As result we obtain a 3D model of the scene ready for the scene exploration (Fig. 5 left). The reconstructed scene is stored on a server as a scene-graph and its relative spheremaps. At run-time, this graph is explored through the system proposed in Sec. 4.3.

4.3 Scene exploration

In order to browse the scene we adopt an image-based navigation system on a client/server architecture. The server is a standard *http* server (see Sec. 5) hosting the scene-graph and the images, whilst the client is an WebGL-based interactive viewer, implementing an assisted interaction paradigm illustrated in Figure 5 (center and right). Similarly to other approaches already proven effective such as [9], the viewer operates in two modes: either it shows a panoramic

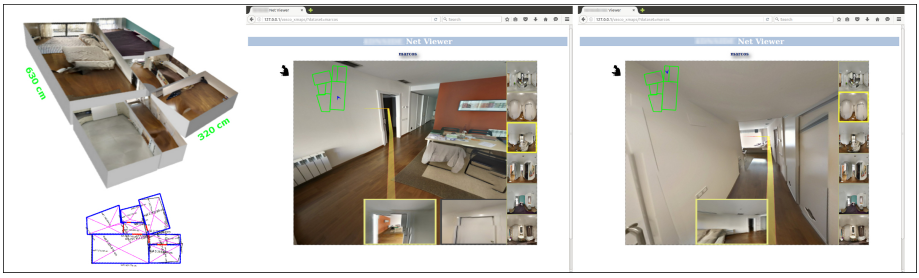

Fig. 5. Left: top view of the 3D scene reconstructed by our system (T3 Tab 1). Center and right: screenshots of interactive browsing of the scene through the Viewer module. The client/server architecture has been defined to achieve a platform independent navigation system running on standard web browsers.

image or it shows the transition from one room to the next. A common way to show equirectangular images consists of first converting the them to cubemaps and then using the cubemap texture capabilities of graphics API to render the image. Instead, we use a simple fragment shader that, for each fragment, takes the direction of the corresponding viewing ray and samples the equirectangular image accordingly. The interaction consists of simply dragging to change view orientation and pinching (or using mouse wheel for non touch screens) to zoom in/out. The right thumbnail bar shows instead all available rooms, and the bottom context-sensitive thumbnail bar shows selected rooms reachable from the current position. Panoramas in the *path bar* correspond to paths leaving the current room (Fig. 5 right, yellow), where the panorama icons are ordered accordingly to the angle between the current view direction and the path direction, so as to always center the rooms bar on the path most aligned with the current view. Such paths are obtained from the reconstruction step as the piecewise line segment connecting two adjacent room centers and passing through the doors. Furthermore a contextual 2D map (Fig. 5 right, green map) of the scene is provided to improve the localization of the viewer in the scene. Unlike in other previous methods [9, 25], the transition between panoramas is not a video, that is, our viewer is not purely image based. Instead, thanks to the structured model reconstructed by our system, we create a textured mesh by projecting the panoramas on the 3D model. Since our geometric model only consists of the room's boundary (floor, ceiling and walls) it is clear that projecting the images that also include elements within the room (tables, chairs etc.) creates a model which is only correct when seen from the center of projection while the error become apparent when moving away. However, it must be considered that, during a transition, the viewing direction and the direction of movement coincide, resulting in a coherent representation. The main advantage of these solutions is that the network bandwidth occupancy is reduced (e.g. with respect to [9, 25]), as well as the lag due to video buffering during the movement between rooms, resulting in real-time performance even on low-powered mobile devices or poor network coverage. Summarizing, the tool exploits the specific characteristics of the environment to create a constrained navigation model that does not hin-

der the possibility of smoothly inspect the scene, providing at the same time an aided user interface that any user can master and so avoiding that the user "feels lost" in the 3D scene.

5 Results and discussion

To demonstrate the effectiveness of our approach we present its application on real-world cases. As proof-of-concept we employed our method as an assistive technology in the context of security assessment and management of critical buildings. We tested the system with a group of 60 users including different profiles (police, crisis managers and trainers, first responders, fire services), specifically none of them are computer experts, CAD modelers,etc. Such implementation is integrated in to a more general crisis management framework (*reference removed for blind review*) assisting crisis managers and first responders to visually define and assess concepts and measures for protecting buildings.

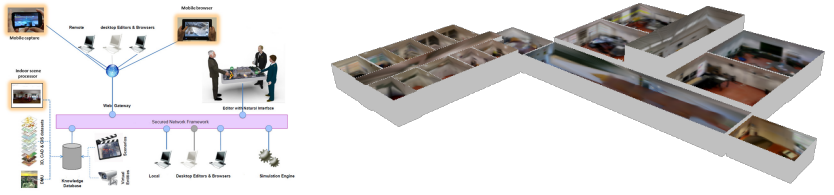

Fig. 6. Left: The proposed pipeline is integrated in the main framework through a specialized local area network (illustrated above). Right: Example of a 19-rooms environment (T1 Tab 1) employed as training scenario.

The *Scene Analysis* module is interconnected with the main *server* by a local area network (Fig. 6), whereas the *Mobile Capture* module and the *Viewer* module are available through a web gateway. The capture tool is implemented as a remote Android application (Android 4.4 or higher compatible) to acquire the indoor environment, the *Scene Analysis* is a stand-alone desktop application to process and upload the data to an *Apache2* web server, and the *Viewer* is a platform independent viewer written in *JavaScript* using *WebGL* and *HTML5*.

The acquisition, reconstruction and visualization tasks were designed in collaboration with real crisis management experts [7], evaluating the system in terms of usability in a real-world application. The scene acquisition was performed by users (i.e., first responders in our tests) equipped with portable devices, such as touchscreen tablets, and running the *Mobile Capture* module. We asked them to rate the usability of capturing the indoor environment with the provided mobile device (HTC One M8) on a scale from 1 to 5 (very difficult, difficult, moderate, easy, very easy). 60% of the users found it very easy to capture the indoor environment, whilst 40% found it to be easy (Fig. 7 left). Managers and trainers instead had access to Scene Server through the *Scene Analysis* module to analyze the reconstructed scene. We asked to rate their satisfaction with

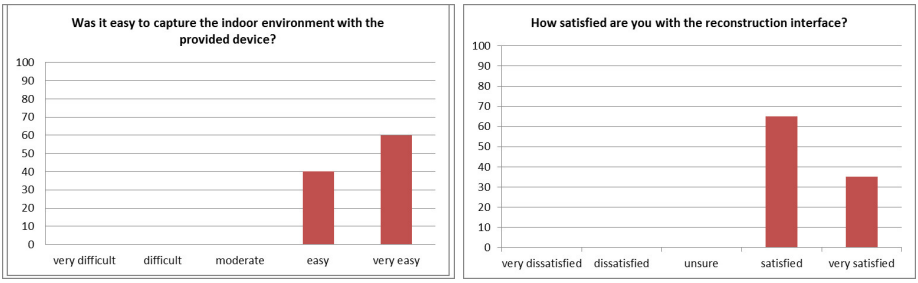

Fig. 7. User score in terms of usability respectively of the capture tool and the reconstruction tool. The system has been tested by a selected end user group of about 60 users.Although a minimal manual interaction was needed in the reconstruction step, the users were very satisfied of the acquisition and reconstruction tools, especially if compared to the standard alternatives (basically manual measurements).

the indoor reconstruction system using a scale from 1 to 5 (very dissatisfied, dissatisfied, unsure, satisfied, very satisfied). 35% end users declared themselves very satisfied with the indoor reconstruction, whilst 65% of them were satisfied with it (Fig. 7 right).

Scene			Error			Interaction		
Name	**Area**	**Rn**	**Area**	**Length**	**Height**	**Capture**	**Re**	**Editing**
T1	$655m^2$	19	2.5%	11 cm	7 cm	51m45s	3	8m25s
T2	$183m^2$	3	1.1%	15 cm	5 cm	8m20s	1	3m12s
T3	$64m^2$	7	2.3%	12 cm	4 cm	18m35s	1	2m40s

Table 1. Reconstruction statistics. We indicate the floor area and the number ***Rn*** of input panorama images/rooms. In the second columns group we show the **maximum** error measured compared to ground truth for the walls length and room height, after both the automatic and interactive fixing steps. Finally we show the total capture time, the number ***Re*** of rooms needing manual fix after the automatic recognition and the relative editing time.

In Tab. 1 we show the capture and reconstruction statistics for several significant scenes, comparing our system to the ground truth. We assume as ground truth blue-prints (when available) and manual measures. The area error is calculated as the ratio of area incorrectly reconstructed to the total ground truth area, and the wall length and wall height error are the maximum error measured *after* the manual editing step. This error is not correlated with a specific user (i.e. user interaction for the room's shape recovery consists only in to mark room's corners in the image when the automatic detection fails), but is strictly related to the quality of the images stitching. In the last columns group we show as Capture Time the time needed by an user to acquire a complete stitching of each room and to walk from room to room, finding in every new environment a convenient position to capture the spheremap. ***Re*** is instead the number of rooms needing manual interaction (Editing time showed in the last column).

As said above the main cause of failure and error in the reconstruction is the quality of the stitching. In the specific case of dataset T1 and T3 the failure cases

are corridors, whose stitching is highly distorted. Despite manual intervention a tangible scale error remains. Differently in dataset T2 the automatic recovery didn't really failed in any room, but since the scenario was an ancient building with very highly textured walls, the user had to manually define which was the real ceiling level relevant for the reconstruction. In Fig. 6 right we show the reconstruction of the 19 rooms environment, indicated as T1 in Tab. 1, seen from an above view. The returned model is metrically scaled in real-world dimensions, included the different heights of each room correctly represented. To test the *Viewer module* we asked to a group of end-users NOT involved in the capture and reconstruction tasks, to explore a new acquired environment, employing mobile devices (smartphones and tablet, not only Android) or generic internet browsers (Explorer, Firefox, Chrome) running on commodity PCs and laptops. Users have found the interface comfortable and intuitive, and the first responders category in particular quickly familiarized themselves with the application thanks to the similarity with popular web-based 3D map navigators. We connect the devices through a *wireless-N* local lan and through 3G mobile connection, achieving in all cases a frame rate exceeding 50fps. Differently to approaches based of pre-computed video sequences [9], where the frame rate during video transitions drops down to about 20 fps due to video decompression, our system keep the maximum frame rate also during moving between rooms, thanks to the real-time rendering performed on the simplified model. In Fig. 5 we present the 3D model recovered for the T3 dataset (Tab 1) and some live screenshots of the interactive browsing through the *Viewer module*. In terms of assistance in designing security, almost all participants stated that the proposed framework adds value to the general process of designing building security, mainly for the capability of being able to visualize a model reconstructed from real data and not replicated by a 3D modeler. More specifically, users are interested in the possibility of designing and testing real path planning and evacuation routes, as well as in being able to simulate the introduction of more sensors in the environment or to incorporate the tools into daily use, integrating it with the CCTV system of a building for example.

6 Conclusions

We presented an image-based system to acquire, reconstruct and explore indoor scenes, starting from data acquired with mobile devices. Combining mobile and web technologies this system can be an effective tool assisting different users to reconstruct and share realistic models of buildings, without the need of any particular 3D expert skill. The resulting model contains enough geometry to support common simulation techniques, and enough visual information to effectively support visual navigation and location recognition. Since the returned scene is already simplified and structured, we exploit such structuring to support a real-time browsing of the acquired scene through a fast interactive WebGL rendering system, running on common mobile devices. For the future we plan to exploit multiple omnidirectional point-of-views, such as the video output of

modern mobile 360 camera, to improve scene understanding and/or to support assisted navigation.

Acknowledgements

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 under projects VIGEC and Vis&VideoLab.

References

1. Ariani, A., Redmond, S.J., Chang, D., Lovell, N.H.: Simulation of a smart home environment. In: Instrumentation, Communications, Information Technology, and Biomedical Engineering (ICICI-BME), 2013 3rd International Conference on. pp. 27–32 (Nov 2013)
2. Bermudez-Cameo, J., Puig, L., Guerrero, J.: Hypercatadioptric line images for 3d orientation and image rectification. Robotics and Autonomous Systems 60(6), 755 – 768 (2012)
3. Brivio, P., Benedetti, L., Tarini, M., Ponchio, F., Cignoni, P., Scopigno, R.: Photocloud: interactive remote exploration of large 2d-3d datasets. IEEE Computer Graphics and Applications 33(2), 86–96 (2013)
4. Cabral, R., Furukawa, Y.: Piecewise planar and compact floorplan reconstruction from images. In: Proc. CVPR. pp. 628–635 (2014)
5. Clarke, D., Duimering, P.R.: How computer gamers experience the game situation: A behavioral study. Comput. Entertain. 4(3) (Jul 2006)
6. Coughlan, J.M., Yuille, A.L.: Manhattan world: Compass direction from a single image by bayesian inference. In: Proc. ICCV. vol. 2, pp. 941–947 (1999)
7. Crisis Plan BV: Crisis management experts, http://www.crisisplan.nl/
8. Dev, K., Lau, M.: Democratizing digital content creation using mobile devices with inbuilt sensors. Computer Graphics and Applications 35(1), 84–94 (Jan 2015)
9. Di Benedetto, M., Ganovelli, F., Balsa Rodriguez, M., Jaspe Villanueva, A., Scopigno, R., Gobbetti, E.: Exploremaps: Efficient construction and ubiquitous exploration of panoramic view graphs of complex 3d environments. Computer Graphics Forum 33(2) (2014), proc. Eurographics 2014
10. Flint, A., Mei, C., Murray, D., Reid, I.: A dynamic programming approach to reconstructing building interiors. In: Proc. ECCV. pp. 394–407. Springer (2010)
11. Furukawa, Y., Curless, B., Seitz, S.M., Szeliski, R.: Reconstructing building interiors from images. In: Proc. ICCV (2009)
12. Geyer, C., Daniilidis, K.: A unifying theory for central panoramic systems and practical implications. In: Proc. ECCV. pp. 445–461 (2000)
13. Goesele, M., Ackermann, J., Fuhrmann, S., Haubold, C., Klowsky, R., Steedly, D., Szeliski, R.: Ambient point clouds for view interpolation. ACM Trans. Graph. 29(4), 95:1–95:6 (Jul 2010)
14. Goetz, M., Zipf, A.: Open issues in bringing 3d to location based services (lbs): a review focusing on 3d data streaming and 3d indoor navigation. Remote Sensing and Spatial Information Sciences pp. 38–4 (2010)

15. Guest, J., Eaglin, T., Subramanian, K., Ribarsky, W.: Interactive analysis and visualization of situationally aware building evacuations. Information Visualization (2014)
16. Helal, S., Mann, W., El-Zabadani, H., King, J., Kaddoura, Y., Jansen, E.: The gator tech smart house: a programmable pervasive space. Computer 38(3), 50–60 (March 2005)
17. Hoiem, D., Efros, A.A., Hebert, M.: Geometric context from a single image. In: Proc. ICCV. vol. 1, pp. 654–661. IEEE (2005)
18. Jahromi, Z.F., Rajabzadeh, A., Manashty, A.R.: A multi-purpose scenario-based simulator for smart house environments. (IJCSIS) International Journal of Computer Science and Information Security 9(1), 13–19 (Jan 2011), http://dx.doi.org/10.1007/s00779-014-0813-0
19. Mura, C., Mattausch, O., Gobbetti, A.J.E., Pajarola, R.: Automatic room detection and reconstruction in cluttered indoor environments with complex room layouts. Computers & Graphics (2014)
20. Nguyen, T.V., Kim, J.G., Choi, D.: Iss: The interactive smart home simulator. In: Advanced Communication Technology, 2009. ICACT 2009. 11th International Conference on. vol. 03, pp. 1828–1833 (Feb 2009)
21. O'Neill, E., Klepal, M., Lewis, D., O'Donnell, T., O'Sullivan, D., Pesch, D.: A testbed for evaluating human interaction with ubiquitous computing environments. In: First International Conference on Testbeds and Research Infrastructures for the DEvelopment of NeTworks and COMmunities. pp. 60–69 (Feb 2005)
22. Pintore, G., Garro, V., Ganovelli, F., Agus, M., Gobbetti, E.: Omnidirectional image capture on mobile devices for fast automatic generation of 2.5D indoor maps. In: Proc. IEEE Winter Conference on Applications of Computer Vision (WACV) (February 2016)
23. Pintore, G., Gobbetti, E.: Effective mobile mapping of multi-room indoor structures. The Visual Computer 30 (2014), proc. CGI 2014
24. Pintore, G., Gobbetti, E., Ganovelli, F., Brivio, P.: 3dnsite: A networked interactive 3d visualization system to simplify location recognition in crisis management. In: Proc. ACM Web3D International Symposium. pp. 59–67. ACM Press, New York, NY, USA (August 2012)
25. Sankar, A., Seitz, S.: Capturing indoor scenes with smartphones. In: Proc. UIST. pp. 403–412. ACM, New York, NY, USA (2012)
26. Seitz, S.M., Curless, B., Diebel, J., Scharstein, D., Szeliski, R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In: Proc. CVPR. vol. 1, pp. 519–528 (2006)
27. Snavely, N., Seitz, S.M., Szeliski, R.: Photo tourism: exploring photo collections in 3D. ACM Trans. Graph. 25(3), 835–846 (Jul 2006)
28. Wilson, C., Hargreaves, T., Hauxwell-Baldwin, R.: Smart homes and their users: A systematic analysis and key challenges. Personal Ubiquitous Comput. 19(2), 463–476 (Feb 2015), http://dx.doi.org/10.1007/s00779-014-0813-0
29. Xiong, X., Adan, A., Akinci, B., Huber, D.: Automatic creation of semantically rich 3D building models from laser scanner data. Automation in Construction 31(0), 325 – 337 (2013)