

Crime Scene Reconstruction with RGB-D Sensors

Abdenour Amamra, Yacine Amara, Khalid Boumaza, and Aissa Benayad

Ecole Militaire Polytechnique, Bordj El-Bahri BP 17, Algiers, Algeria

Email: amamra.abdenour@gmail.com

Abstract—Photographic surveying, a fundamental procedure in crime investigation, is typically performed using 2D cameras. Although useful, such cameras remain limited due to the lack of depth information. In this work, we propose a 3D reconstruction solution that leverages the advantages of cheap RGB-D sensors to create a 3D model of the crime scene and to provide the investigator with an interactive crime scenario simulation environment. A structure from motion approach is proposed in order to align the captured point clouds on each other using 3D key points. An iterative refinement and a global optimization algorithm are later adapted for the optimization of the registered 3D model, which is then triangulated before the underlying surface is reconstructed. The resulting model is used for interactive crime investigation and object dynamics simulation. The obtained results show the effectiveness of our solution with a visually appealing rendering, an accurate simulation and a quantitative error of less than 18cm for the $4m \times 4m$ indoor scene. An accompanying video is provided in order to illustrate the processing pipeline¹.

Index Terms—Crime scene modeling, 3D registration, RGB-D sensors, Forensic computing, Interactive investigation.

I. INTRODUCTION

In the world we live today, we hear frequently about crimes and the ways they are resolved. Freezing and preserving the crime scene is certainly one of the most important and delicate initial steps that the police perform upon the arrival to a crime scene. The idea is to conserve the spatial configuration of the objects surrounding the place where they think the crime took place for the purpose of avoiding any contamination.

In most current cases, the investigation is carried out in the simplest manner, that is, with paper, pencil and measurement ribbon, to sketch an illustration of the crime scene showing the position of the victim, as well as the other tools that could have served the criminal. For instance, a trivial improvement of the simple flat drawings can be a 3D representation of the scene allowing a better understanding of the series of events that conducted to the crime. For instance, several computer tools have been used by experts for the creation, the visualization, and sharing of electronic crime data among investigators. A first 3D reconstruction method based on this software is the design of a 3D model of the scene. Then taking pictures and pasting them on the 3D mesh. This method is simple and does not require much experience, however, it fails to preserve the 3D shape and dimensions of objects. Another method of 3D scene reconstruction is a faithful recreation of the scene. The latter requires the mastery of sophisticated 3D modeling tools and the measurements of all the important scene objects. Its advantage is that the resulting model is very close to reality.

¹<https://youtu.be/IYnJSNV7QkI>

Unlike the methods mentioned above, our approach is based on the alignment of several 3D images of the same scene, with different points of view. This alignment is looking for spatial transformations that merge the views into a single globally consistent model. With such an approach, the reconstruction of scenes is not left to the skills and performance of a human actor; rather, it relies on real environmental measurements while taking into account the dynamics of objects. Interestingly, with a coherent 3D representation and a physics engine, physical laws can be simulated (e.g. simulation of a bullet's trajectory). Otherwise, it would be possible to test crime hypothesis without the need for deployment on the physical scene. For this purpose, the Kinect could serve as an affordable 3D scanner that has interesting performance in 3D shape capturing. Indeed, RGB-D cameras capture the color and the geometry of the scene and deliver colored 3D point clouds, but their data still needs a chain of preprocessing and 3D registration as well as surface inference in order to become a useful representation. In our case, these sensors were placed at the center of the crime scene and the forensic police will take care of swiveling it in order to scan the objects of interest.

In the remainder of the paper, we first discuss the literature of crime scene reconstruction and modeling in Section II. Then, we present our approach to 3D reconstruction and interaction in Section III. We validate our findings with several quantitative and qualitative assessments in Section IV. And we conclude with a summary of what we aimed for and what we really achieved, as well as some future perspectives that can enhance and extend the present work in Section V.

II. RELATED WORKS

The attempt to reconstruct crime scenes with 3D reconstruction means finds its origins in [1]; where the authors used a mobile camera in a Structure-from-Motion (SFM) pipeline. Afterward, the authors in [2] presented a general comparison of 3D imaging sensors for criminal investigation. They took into account most of the 3D techniques available at that date. It should be noted, however, that Time of Flight (TOF) technology, the working principle of the Kinect v2, was not considered since the technology has not been widespread until the last decade. The authors in [3] used an alternative dense reconstruction approach directly on video sequences.

Initially used to solve the problem of ego-motion, Simultaneous Localization and Mapping (SLAM) techniques have been a hot topic in robotics for several years. Crime scene reconstruction works had followed through the light of SLAM technology; where the authors of [4] investigated the

utilization of a stereo camera rig in a SLAM based crime scene reconstruction pipeline. Nevertheless, it is worth noting that stereo mapping is a passive technique that reconstructs a cloud of 3D points by matching the corresponding key points found in both RGB images. Since it relies only on the feature points of the environment, RGB imaging techniques are subject to singularities when the images lack distinctive patterns (e.g. uniformly colored surfaces, which potentially characterize indoor scenes; thus, rendering stereo pairing difficult and resulting in a rough and distorted 3D content).

Another approach is to design a 3D model of the scene using 3D modeling software such as 3ds Max [5]. More recent approaches use laser scanners [6] to reconstruct the crime scene at a very high accuracy. An example of the laser scanners that were used for this purpose is FARO S-350 [7]. The latter is dedicated to fast and accurate 3D indoor and outdoor environments. Nonetheless, its high cost and lengthy reconstruction time are two famous shortcomings of such a technology.

In the light of the literature, and in the purpose of building upon the previous contributions, we investigate the utilization of RGB-D data, in an SFM pipeline with a loop closure mechanism and a 3D triangulation and surface reconstruction algorithm for the development of an interactive crime investigation solution.

III. 3D CRIME SCENE RECONSTRUCTION

In what follows, we present our solution to 3D crime scene reconstruction and interactive investigation.

A. Solution Overview

Figure 1a illustrates the process followed to complete our work. As illustrates the diagram, our solution is essentially divided into three steps: a *data acquisition*, where the point clouds are delivered by the Kinect v2 RGB-D sensor. The second step is the *prepossessing* of the point clouds delivered by the sensor. The latter includes: down-sampling, filtering, and key points extraction and description. The last step, which is the most important, is the *3D alignment and reconstruction* that results in a coherent holistic 3D model for investigation purposes.

B. Data Acquisition

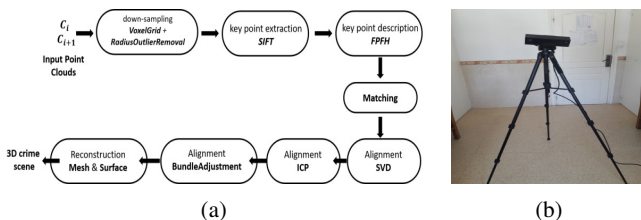


Fig. 1: Crime scene reconstruction setup. (a) Reconstruction pipeline, (b) RGB-D data acquisition setup

This phase involves collecting data from the scene with the Kinect, with the consideration that two successive images

overlap each other. To this end, and as the Kinect sees only the scene within its viewport, we place it at the center of the scene (see Figure 1b) and we have it rotated at small regular angles. We can, therefore, acquire images at multiple views during a complete turn for later reconstruction.

Despite the use of the rotating acquisition procedure, parts of the scene remain occluded in the captured views. In fact, these regions are not visible to the camera as shows Figure 2b. Such a phenomenon can be overcome by adapting the scans to cover the holes. At this level, we focus only on the acquisition of visible parts, because the missing parts can be treated separately after the reconstruction finishes.

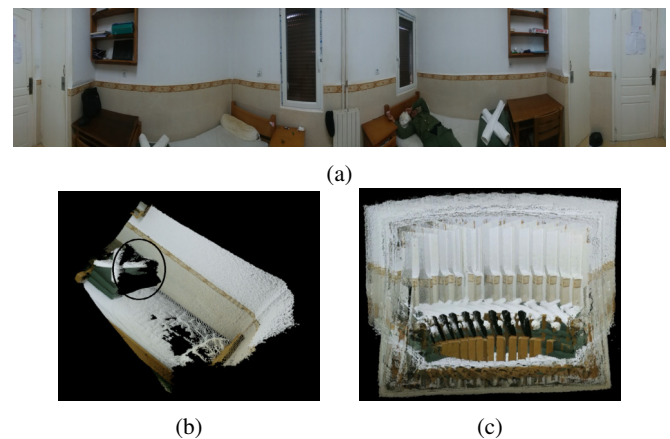


Fig. 2: RGB-D images captured from the experimental $4m \times 4m$ crime scene. (a) Panoramic view of the crime scene, (b) Point cloud, with hidden regions, obtained from the captured view, (c) Projection of all the views in the same reference frame.

C. Filtering and down-sampling

In order to obtain a decent 3D model of the scene, and in the sake of faster 3D mesh creation, it is important to down-sample the raw point cloud. The down-sampling technique that we adapted is based on the Voxelgrid method [8]. After a consistent sampling, we obtain a less dense point cloud that preserves the underlying geometry without redundancy. The second preprocessing to be carried out is noise and outlier points elimination. For this purpose, we adapted the Radius Remove Outlier filter [9]. The latter smoothes the 3D points and removes isolated points.

D. Alignment and 3D reconstruction

This section is essential to our contribution, the objective here is the alignment of the different views in order to obtain a coherent 3D scene. Our approach to point cloud alignment is cumulative pairwise. In other words, we align a pair of point sets at a time, then we carry on with the following pair, and so on; until the last view (typically overlapping the first one, as we rotated the camera in order to scan the whole scene from the center).

Technically the alignment of the point sets results from the computation of a 4×4 transformation matrix T . The latter embeds a 3×3 rotation matrix and a 3×1 translation vector. We add to this matrix the so-called homogeneous coordinates for the simplicity of computation to obtain the 4×4 transformation matrix.

$$T = \begin{bmatrix} r_{00} & r_{01} & r_{02} & t_0 \\ r_{10} & r_{11} & r_{12} & t_1 \\ r_{20} & r_{21} & r_{22} & t_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

For example, in order to align three point clouds A_1, A_2, A_3 together, one first aligns the second cloud A_2 with the first A_1 , where the result after minimization is the transformation $T_{1,2}$. We obtain a global intermediate result $G_{1,2} = A_1 \cup T_{1,2} * A_2$. Then, we align the third cloud A_3 with the already obtained global result $G_{1,2}$ to obtain the final result $G_{1,2,3} = G_{1,2} \cup T_{1,2,3} * A_3$.

In order to perform point cloud alignment, we propose the following scheme:

1) Initial alignment

This phase aims to find the transformation between the different views respective to the acquired point clouds (see Figure 2c) after the matching of key points. The initial alignment of two point clouds (typically called source Sr and target Tr , where $Sr_{aligned} = T * Sr$) begins with the extraction of characteristic points; then the computation of the descriptors respective to these points, followed by the estimation of correspondences. After the rejection of bad matches, we estimate the transformation that aligns the sets of points and we apply it on the source point set. In our work, we initially adopted the matrix Singular Value Decomposition (SVD) [10] in order to estimate an initial guess of the underlying transform. This preliminary result is prone to error since it is computed fast without any sophisticated refinement (see Figure 3a). Nevertheless, it serves as a desirable initialization for the subsequent Iterative Closest Point (ICP) [10] alignment algorithm which gets frequently trapped in local minima.

2) Iterative refinement with ICP

In most cases, the initial alignment does not give sufficiently accurate transformation. Hence, our goal now is to refine as much as possible the already obtained transformation with the ICP algorithm. The latter finds first the closest points in both clouds of points (correspondence estimation); then it proceeds through the estimation of a transformation that best aligns the matched correspondences (alignment). These two steps (correspondence, alignment) are repeated until reaching a termination criterion (typically, an error threshold or a maximum number of iteration). Figure 3b shows the outcome of applying the ICP algorithm on the SVD result. Although accurate when aligning a few point sets, ICP inherently leads to cumulative small errors due to a large number of views (around 50 views for a small room).

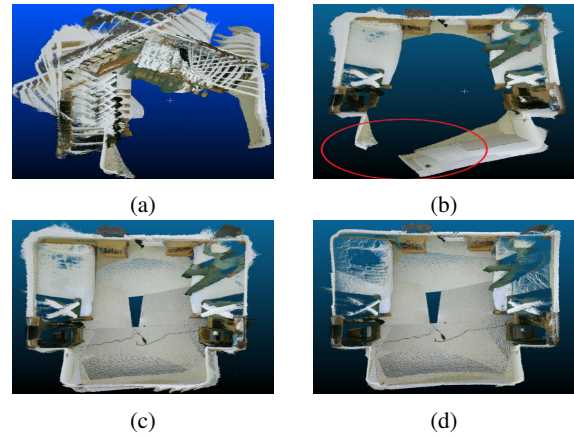


Fig. 3: Point cloud Alignment Results. (a) After initial alignment with SVD, (b) After refinement with ICP, (c) After loop closure, (d) After filtering and segmentation.

The global error between all the aligned 50 views after SVD is about 7m, which is huge for a $4m \times 4m$ room. Indeed, ICP greatly minimizes this error to around 1m, but the accumulation of errors still impinges on the loop closure, due to a misalignment between the first and the last point set, that needs to be dealt with.

E. Loop Closure

As we mentioned earlier, point cloud alignment techniques introduce a non-negligible cumulative error, but the closure between the first and the last point clouds significantly reduced the error to its lowest levels. We noticed that this is not always the case, as when misalignment due to mis-correspondences is encountered, the overall reconstruction is no longer reliable. The reconstruction error increases with distance and misalignment is difficult to correct. Figure 4 illustrates the accumulation of misalignments after the application of SVD and ICP algorithms. This accumulation is more noticed at the corners of the room.

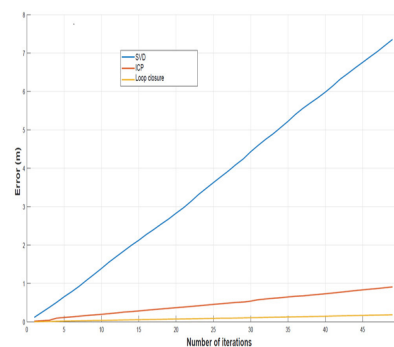


Fig. 4: The evolution of the cumulative error after SVD, ICP, and Loop Closure.

F. Filtering and segmentation

It is important that the point cloud after alignment bears little noise. For instance, the alignment algorithm filters some

Input: N point clouds $C = \bigcup_{i=1}^N C_i$.

Output: Optimal transformation (\hat{T}) that minimizes the global misalignment (f) between the C_i .

- 1) **Calculate the correspondences** between a given pair of clouds S, D ; such that $(S, D) \in C \times C$ and $S \neq D$.
 - a) Initialize $S' = \emptyset, D' = \emptyset$.
 - b) For each point $p \in S; \forall q \in D$:
 - i) Calculate the distances between p and q .
 - ii) The point q of minimum distance corresponds to p .
 - iii) Remove p from S , and it to S' .
 - iv) Remove q from D , and it to D' .
 - c) Choose two subsets S'' and D'' of S' and D' , respectively, such that the elements of S'' and D'' are those of S' and D' with a distance below a given threshold.
- 2) **Estimate T** :

$$\hat{T} = \arg \min_T \left(\sum_{\forall S \neq D \in C} \|D - T * S\|^2 \right)$$

- 3) **Repeat 1 and 2** : until the following criterion is satisfied ($f(\hat{T})$ is the global alignment error between all the N views after applying the transformation \hat{T}) :

$$\|f(\hat{T})\|^2 \leq \epsilon$$

Algorithm 1: Loop closure Algorithm

of the noise but, if the cloud carries a lot of points not belonging to the model, the result of reconstruction may not be convincing to the investigator. After filtering our resulting point cloud with the Region Growing Segmentation filter [9] we obtained the result in Figure 3d.

G. 3D triangulation and surface reconstruction

This step consists in creating facets from the point cloud by connecting the non-ordered points to each other in a triangular topology. To this end, we first apply Delaunay triangulation algorithm [9] for the creation of a smooth mesh. Figures 5a, 5b show our scene after surface reconstruction.

When we zoom in on the different parts of the scene (Figure 5b), we notice that surface reconstruction is generally of good quality except in some minor regions. These regions are typically characterized by a low density of 3D points resulting in a poor quality triangulation. To solve this problem, we captured 2D colored images during the acquisition phase, which will subsequently be used to fill the holes.

IV. RESULTS AND DISCUSSION

In this section, we analyze the performance of our Loop Closure algorithm applied to improve the performance of 3D point cloud alignment. In order to demonstrate the intake of Loop Closure, we focus only on visual results and the variation of the cumulative error after the application of the three alignment algorithms. Figures 3c, 3d illustrate the final

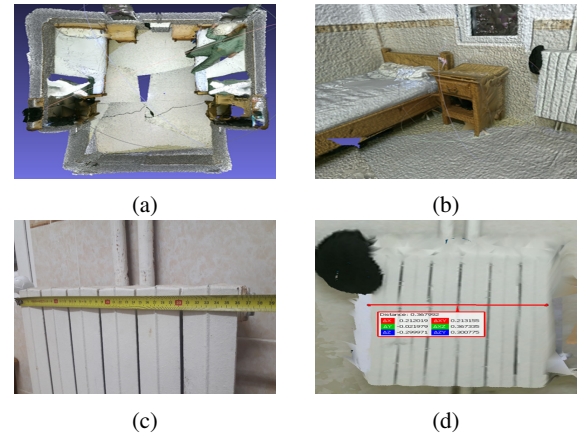


Fig. 5: Result After 3D Reconstruction. (a) Scene overview, (b) Zoom in the corner, (c) Real scene measurement, (d) 3D model measurement.

result of our scene after the alignment of all the views, we can see with the naked eye that the scene is closed, and the algorithm enhances the preceding result. We can also say that the alignment mean squared error has decreased considerably, it results in a misalignment of 18cm which is small given the 4m × 4m area of the room. The graph in Figure 4 illustrates this improvement.

A. Implementation

The implementation of our proposed solution was achieved in C++ using the PCL library², and Matlab 2017b on Windows 10 operating system. The different tests were carried out on a machine equipped with an Intel Core i5 processor of 2.30GHz and 6GB Ram.

In order to quantify the accuracy of our reconstruction, we compare the actual size and dimensions of scene objects to those obtained in the reconstructed model. This assessment is performed through cloud compare software³.

B. Cloud Compare

This utility permits us the measurements of distances between 3D points. To make this comparison, we took real measurements of some objects in the crime scene, and we measured their respective reconstructed counterparts using Cloud Compare as shown in Figure 5c, 5d. Table I contains some measured dimensions of objects that were present in the crime scene. We notice clearly that our model gives enough accurate results, which were validated by investigators. These measurements mean that there is little overall difference between our model and the actual scene, where to 98% both measurements were similar. In addition, measurement error is more important in the larger objects due to the decreasing accuracy of the depth map delivered by the sensor.

²<http://www.pointclouds.org>

³<https://www.danielgm.net/cc/>

Objects	Measurements in centimeter (cm)		
	Real	3D Model	Error
Room width	384	400.4	16.4
Nightstand	46	46.6	0.6
Radiator	36	36.8	0.8
Chair	43	43.9	0.9
Cupboard door	74.5	75.7	1.2
Entrance door	99	99.2	0.2
Shelf	80.5	81	0.5
Table	85	86.1	1.1
Window	58	58.6	0.6
Bed width	97	97.4	0.4
Bed length	204	205.5	1.5
Electric socket	39	39.2	0.2

TABLE I: Comparison between actual measurements and model measurements.

C. Crime scenario

A good 3D reconstruction of the scene helps the investigator in the investigation, we simulated a crime scene that can enrich our work. This scenario is intended to reproduce the hypothesis prior to the action of committing the crime. To this end, we used Unity software ⁴, which is a multi-platform game engine. This software allows us to introduce virtual objects into our scene and then to make animations and to create scenarios. In our case, we used two virtual persons (victim and criminal) as shows Figure 6. We programmed these characters in order to obtain a scenario as close as possible to reality.

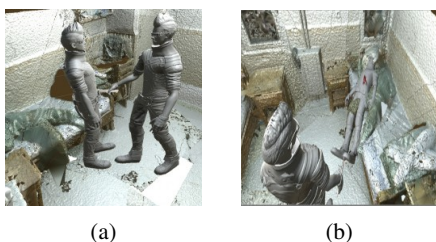


Fig. 6: Unity 3D simulated crime scenario: (a) Before, (b) After the crime.

D. Technical issues

We noticed during the deployment of our solution several technical problems. The first one concerns the stage of data acquisition, the Kinect sensor uses infrared rays to infer the depth of objects. Nevertheless, there are some hidden parts that could not be captured by the sensor. Another problem was encountered in the global alignment process, this problem is due to the accumulation of the errors. Correspondence estimation algorithms that are based on the feature descriptor are not optimized, which brings to bad matches, all of this leads to misalignment. The surface reconstruction stage generally gives good results, with the exception of the hidden or sparse regions. The last problem was noticed in Unity 3D during the simulation of crime scenarios, as the game engine accepts only three scene formats (.obj, .fbx, or .unity), we were obliged to

convert our model and to lose some scene’s color and geometry consistency.

V. CONCLUSION

We presented an RGB-D solution for accurate crime scene reconstruction and interactive scenario simulation. We leverage the potential of the cheap Kinect v2 TOF sensor in order to scan the crime space. Based on a set of key points we applied a preliminary alignment of the different views with SVD optimization. The latter is known for its speed but poor accuracy. However, it delivers a good guess for the subsequent iterative refinement algorithm (ICP). Afterward, we addressed the loop closure problem with a bundle adjustment algorithm.

Once all the views registered in the same reference frame, we proceeded through the triangulation and surface reconstruction. The resulting model is used for interactive crime investigation and object dynamics simulation. The obtained results show the effectiveness of our solution and its adequacy to the context being treated. We demonstrated the performance of our finding on a real scene. An accompanying video illustrates the whole chain of processing is available ⁵.

As perspectives, we aim to extend the capabilities of the proposed solution to work outdoors. Regarding the external reconstruction performance of Kinect v2, we need to address the sensitivity to sunlight. Moreover, since our solution is better suited for offline processing, it would be interesting to investigate possible acceleration techniques for rapid on site reconstruction through the utilization of graphics processors and highly parallel techniques. Finally, the theory behind our work can be exploited for large scale reconstruction as well, after taking into account the required computational burden and the intervention of the different reconstruction elements.

REFERENCES

- [1] Simon Gibson and Toby Howard. “Interactive reconstruction of virtual environments from photographs, with application to scene-of-crime analysis”. In: *Proceedings of the ACM symposium on Virtual reality software and technology*. ACM. 2000, pp. 41–48.
- [2] Giovanna Sansoni, Marco Trebeschi, and Franco Docchio. “State-of-the-art and applications of 3D imaging sensors in industry, cultural heritage, medicine, and criminal investigation”. In: *Sensors* 9.1 (2009), pp. 568–601.
- [3] Erkan Bostanci. “3D reconstruction of crime scenes and design considerations for an interactive investigation tool”. In: *arXiv preprint arXiv:1512.03156* (2015).
- [4] Stephen Se and Piotr Jasiobedzki. “Instant scene modeler for crime scene reconstruction”. In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)-Workshops*. IEEE. 2005, pp. 123–123.

⁵<https://youtu.be/IYnJSNV7QkI>

⁴<http://www.unity.org>

- [5] Trung Kien Dang, Marcel Worring, and The Duy Bui. "A semi-interactive panorama based 3D reconstruction framework for indoor scenes". In: *Computer vision and image understanding* 115.11 (2011), pp. 1516–1524.
- [6] Ursula Buck et al. "Accident or homicide–virtual crime scene reconstruction using 3D methods". In: *Forensic science international* 225.1-3 (2013), pp. 75–84.
- [7] Andreas Georgopoulos and Elisavet Konstantina Stathopoulou. "Data acquisition for 3D geometric recording: state of the art and recent innovations". In: *Heritage and Archaeology in the Digital Age*. Springer, 2017, pp. 1–26.
- [8] Jason Ligon et al. "3D point cloud processing using spin images for object detection". In: *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE, 2018, pp. 731–736.
- [9] A Amamra. "Robust 3D registration and tracking with RGBD sensors". PhD thesis. Cranfield University, 2015.
- [10] Yilin Liu et al. "Curvature feature extraction based ICP points cloud registration method". In: *Optoelectronic Imaging and Multimedia Technology V*. Vol. 10817. International Society for Optics and Photonics, 2018, p. 1081707.