3D MODELLING BY LOW-COST RANGE CAMERA:
SOFTWARE EVALUATION AND COMPARISON

R. Ravanelli [a],[*] L. Lastilla [a, b], M. Crespi [a, b]

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

Commission II

KEY WORDS: Range camera, Occipital Structure Sensor[TM], 3D modelling, software comparison

ABSTRACT:

The aim of this work is to present a comparison among three software applications currently available for the Occipital Structure Sensor[TM]; all these software were developed for collecting 3D models of objects easily and in real-time with this structured light range camera. The SKANECT, itSeez3D and Scanner applications were thus tested: a DUPLO[TM] bricks construction was scanned with the three applications and the obtained models were compared to the model virtually generated with a standard CAD software, which served as reference.
The results demonstrate that all the software applications are generally characterized by the same level of geometric accuracy, which amounts to very few millimetres. However, the itSeez3D software, which requires a payment of $7 to export each model, represents surely the best solution, both from the point of view of the geometric accuracy and, mostly, at the level of the color restitution. On the other hand, Scanner, which is a free software, presents an accuracy comparable to that of itSeez3D. At the same time, though, the colors are often smoothed and not perfectly overlapped to the corresponding part of the model. Lastly, SKANECT is the software that generates the highest number of points, but it has also some issues with the rendering of the colors.

1. INTRODUCTION

Nowadays, the range camera technology is sufficiently ripe to play an important role in the field of close-range 3D modeling, offering a cheap, intuitive and effective tool for collecting the 3D geometry (shape and dimensions in metric units) of objects and environments practically in real-time.

Range cameras are indeed active imaging sensors, low-cost and easy-to use, able to natively measure the distances (depths) of several points with a high frame rate (30 - 60 Hz) and therefore they can be considered, to all intents and purposes, veritable 3D scanners. Furthermore, Simultaneous Localization And Mapping (SLAM) algorithms allow to leverage the depth data and the high frame rate that range cameras offer by continually tracking the position of the moving sensor. In this way, the depth maps captured from different points of view are merged into an overall 3D model automatically and in real-time.

Several range cameras (Intel Real Sense family, Kinect family, Google Tango family, Occipital Structure Sensor, ...) are currently available on the market, thus the decision regarding which sensor and which software to use is not straightforward, especially for not expert users, and it should be carefully evaluated since the choice will affect the quality of the resulting 3D models.

The present work is precisely included in this background: the main goal is to perform a comparison among several software applications currently available to capture 3D models of objects with the Occipital Structure Sensor[TM] (see Figure 1), one of the most promising structured light range camera.

Launched on Kickstarter in September 2013 and raising almost 1.3 millions of dollars in 45 days of campaign, the Structure Sen-

sor has been specifically designed to be the first range camera for mobile devices, thus making this technology easily accessible to a wider and inexpert public. For example, archaeologists may now use the Sructure Sensor for easily and quickly documenting the relevant finds discovered during excavations ((Ravanelli et al., 2017a), (Ravanelli et al., 2017b)), since this device can represent a valid alternative to the often time consuming traditional illustration techniques.

Figure 1: The Structure Sensor and the Apple iPad Air 2 tablet used to perform the tests.

This work is therefore aimed at evaluating the pros and cons of each software in order to give even to inexpert users an indication concerning the choice of the best software for their specific purpose. The following applications for the Structure Sensor were thus tested: SKANECT, itSeez3D and Scanner (and its variants).

SKANECT is developed by ManCTL, a French-American company founded in late 2011 and acquired by Occipital in 2013. The software, which runs on a desktop computer (Windows PC or Mac) connected via cable or wireless to the sensor, can use either the *GPU reconstruction* (NVidia graphics card with CUDA support) or the *CPU reconstruction* mode. The first is very sensitive to the geometry of the object being scanned (flat surfaces can be difficult to reconstruct), whereas the latter can also work with not complex geometry, as long as the object has enough texture

---
[*]Corresponding author.

information. For commercial use, the PRO version has to be purchased ($129, unlimited scans), while the free version is available for a personal and leisure use, with the model export functionality limited, though, to 5000 polygons. The SKANECT version 1.8 was used for this work.

ItSeez3D, developed by Itseez Inc., a Russian company leader in implementing Computer Vision algorithms recently acquired by Intel, runs directly on the tablet and requires a post processing step in the cloud. The tracking process uses both geometry and color information; the high resolution texture option can be enabled to prevent the downsampling of the color images to three megapixels, but it obviously slows the uploading of the final 3D model on the cloud. To export the models via mail in .ply or .obj formats, it is necessary to pay $7 per model (free individual subscription). The itSeez3D version 4.3 was used for this work.

Finally, the Scanner application is an integral part of the Structure Software Development Kit (SDK) provided by Occipital: it is free, the source code is available in the form of a sample and can be customized by the developers according to the requirements of the specific use case. The scanning can be performed using either the *Old Tracker* mode, that only leverages geometry information, or the *New Tracker* mode, that also exploits color data to keep track of the object being scanned. The texture quality of the final model can be controlled by enabling the *Low Resolution* or *High Resolution* options. Three variants of the Scanner application were considered in this work: the first one is the default sample included in the version 0.5.5 of the Structure SDK and using a depth map with a resolution of 320×240 pixels; the second one is the result of a customization of the original source code included in the version 0.5.5 of the Structure SDK, specifically modified to use a depth map of 640×480 resolution; the third one is the default sample included in the version 0.6.2 of the Structure SDK and uses a depth map with a resolution of 320×240 pixels.

2. EXPERIMENTAL SETUP

In order to evaluate the quality of the 3D geometry reconstruction carried out by the three different Structure Sensor applications, the 3D model of a DUPLO™ brick construction (Figure 2) was virtually generated with a standard CAD software after having measured its dimensions with a vernier caliper (accuracy of 0.1 mm). In this way, the DUPLO model served as reference and was compared to the 3D models obtained by scanning the correspondent real brick construction with the Structure Sensor by means of the three tested software. Precision and accuracy of the three applications were thus evaluated in terms of signed distances (positive inside and negative outside the reference mesh surface) of the 3D model points from the reference mesh.

Before getting the scan started, the Structure Sensor was calibrated using the Calibrator application provided by Occipital, in this way all the applications shared the same calibration parameters. In fact, the Structure Sensor does not have its own color camera and therefore the object texture is captured by the camera of the device to which it is connected. Hence, the geometry and color are collected from two different points of view and, in order to accurately overlap the 3D data with the corresponding texture, it is necessary to estimate the parameters of the geometric relationship between the Structure Sensor and the camera of the device.

The scanning process was carried out by connecting the Structure Sensor to an Apple iPad Air 2 tablet (see Figure 1), with the high
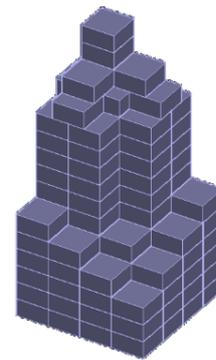

Figure 2: Reference model of the DUPLO™ brick construction.

resolution texture option always enabled for both itSeez3D and Scanner applications.

The scanning took place in an artificially illuminated room, even though the enlightenment was not perfectly uniform. The DUPLO brick construction was put on a table covered with sheets of chequered paper, in order to avoid any reflections of the table and because the color-aided tracking process (itSeez3D, Scanner *New Tracker* mode, SKANECT *CPU reconstruction*) benefits from the presence of surfaces rich in texture near the object to scan.

Moreover, it is worth noting that, during the scanning with the SKANECT software (*GPU reconstruction*), several bricks of different shapes were put around the DUPLO construction, in order to create a more complex scene geometry, useful to the GPU reconstruction.

The scanning procedure required a few minutes for each application, allowing to produce 3D models practically in real-time, with the exception of itSeez3D, which took from 5 to 15 minutes to generate high quality models with the post processing step performed on the cloud.

3. RESULTS

As soon as the 3D models were generated with the three applications (see Figure 3), they were exported in the .OBJ format and imported on the CloudCompare (Girardeau-Montaut, 2017) software, together with the reference mesh. In order to measure the distance between every model generated by the Structure Sensor and the reference, it is indeed necessary to overlap the two models with high precision. The meshes were therefore registered through the Iterative Closest Point (ICP) algorithm (Best and McKay, 1992) implemented in CloudCompare. In particular, the parameters of the roto-translation and the scale were estimated, but only the first ones were used to align the models, since the scale was not significant.

Once the registration was completed, mean and standard deviation of the distances were automatically calculated by CloudCompare: from an operational point of view, for each point of the compared model, CloudCompare simply searches the nearest triangle in the reference mesh and then computes the correspondent distance. The results are summarized in Table 1, which reports the mean and the standard deviation of distances and the number of points of each model for every considered software application.

At a first sight, it is possible to notice how accuracy and precision, whose values amount generally to very few millimeters, do not
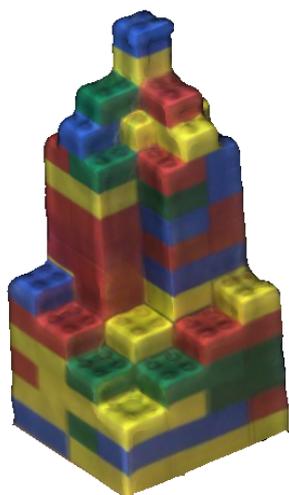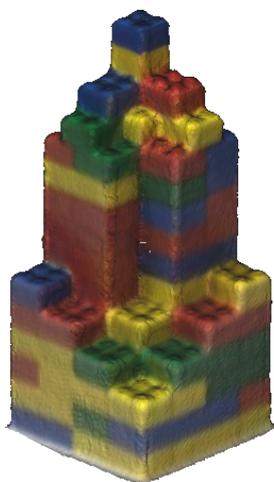
(a) Scanner – SDK version 0.5.5.

(b) Scanner – SDK version 0.6.2, *New Tracker*.

(c) Scanner – modified SDK version 0.5.5, *New Tracker*.

(d) itSeez3D.

(e) SKANECT – *CPU reconstruction*.

(f) SKANECT – *GPU reconstruction*.

Figure 3: Some of the obtained models.

Software application	Tracking mode	Mean [mm]	Std. Dev. [mm]	Number of points [-]
Scanner 0.5.5 SDK Res. 320 × 240	*New Tracker*	0	3	24151
Scanner 0.6.2 SDK Res. 320 × 240	*New Tracker*	0	2	20853
	New Tracker	0	2	20853
	New Tracker	0	2	20798
	New Tracker	0	2	22871
Scanner 0.6.2 SDK Res. 320 × 240	*Old Tracker*	0	2	21553
Scanner 0.5.5 SDK Res. 640 × 480	*New Tracker*	1	2	21218
	New Tracker	0	2	20332
Scanner 0.5.5 SDK Res. 640 × 480	*Old Tracker*	0	2	20913
itSeez3D	-	0	1	25123
	-	0	1	25004
SKANECT	*CPU*	1	1	174957
	CPU	1	1	324511
	GPU	-1	3	323731
	GPU	-1	3	366102

Table 1: Distance statistics of the models.

differ significantly in the models, and this behavior obviously favors a free application like the Scanner by Occipital. Even though the number of models produced with Scanner is bigger than the one of the remaining two software, the samples are sufficiently numerous to prove that, for example, itSeez3D models, sharing the same values of accuracy and precision, show the best results from this perspective, or, on the other hand, that *GPU reconstruction* of the SKANECT software shows lack of precision. This particular behavior of the SKANECT software is probably caused by the fact that the surface of the DUPLO construction does not have enough geometric details, at least in three (flat walls) of its four sides.

In general, from a geometrical point of view, all the software applications showed some difficulties in reconstructing details as the studs, but this problem is to attribute to range camera technology itself.

In particular, not all the variants of the Scanner application guarantee satisfactory results, due to issues as undesired parts inclusions, cut off of specific parts of the object or wrong application of the texture on the mesh. Indeed, an error was found in models generated with the version 0.6.2 of the Scanner, where the lowest part of the DUPLO construction was always systematically not reconstructed in the final 3D models (see Figure 3(b)). Furthermore, the coloring strategy implemented in the software seems to smooth excessively the texture details and, occasionally, the color is not perfectly aligned to the 3D geometry on some parts of the models, especially when using the *Old Tracker* mode. Finally, the increased depth resolution of the modified version does not produce the expected rise in the final number of the model points. Anyway, it is worth to reference to the several advantages of the Scanner application: it is particularly time-saving and free, and, in addition, it is an integral part of the Structure SDK and produces models geometrically comparable to the other software.

As regards itSeez3D, the main advantage and the real strong point of the application is, without any doubts, the texture restitution, which is superior to the other applications taken into account. In addition to this, analyzing the results of the few models collected (thus hoping for further and more detailed experiments), it is possible to affirm that itSeez3D provides also the best geometrical results, as regards accuracy and precision.

For its part, SKANECT is surely the best application to reproduce in detail the geometry of the object, being able to generate dense point clouds with a number of points even ten times the number of any other models; thus this software is able to model the geometry of the bricks in a more detailed way. Nonetheless, in respect of the payment necessary to purchase the PRO version of the application, it suffers from some issues as the creation of points inside the model and a low quality texture reconstruction (the new version of SKANECT claims to supply a better texture restitution).

4. CONCLUSIONS AND FUTURE WORK

In this work, three different scanning applications were tested and compared: SKANECT, itSeez3D and Scanner were used to reconstruct the 3D model of a DUPLO brick construction with the Structure Sensor.

By analysing the obtained results, it is possible to notice that all the software applications are characterized by the same level of geometric accuracy, which amounts generally to very few millimeters. Anyway, considering that both the 3D geometry and the texture information are necessary to provide a complete description of an object, the itSeez3D software is surely the best solution, both from the point of view of the geometric accuracy and, mostly, at the level of the color restitution. However a payment of $7 is required to export, and thus to use effectively, each model produced with this application. On the other hand, Scanner is a free software and shows an accuracy comparable to that of itSeez3D. Therefore, it may be preferable when the economic budget is limited, without worrying to lose too much 3D quality in the final models. At the same time, though, the colors are not fully rendered and not perfectly overlapped to the corresponding part of the model. Lastly, SKANECT is the software that generates the highest number of points, but it has also some issues with the rendering of the colors.

Finally, as a future work, it would be worthy testing the three applications with objects of different size and characterized by more complex geometries.

REFERENCES

Best, P. J. and McKay, N. D., 1992. A Method for Registration of 3-D Shapes. IEEE Transactions on pattern analysis and machine intelligence 14(2), pp. 239–256.

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

Ravanelli, R., Di Rita, M., Nascetti, A., Crespi, M., Nigro, L., Montanari, D. and Spagnoli, F., 2017a. Penguin 3.0-Capturing Small Finds in 3D. Mediterranean Archaeology and Archaeometry 17(2), pp. 49–56.

Ravanelli, R., Nascetti, A., Di Rita, M., Nigro, L., Montanari, D., Spagnoli, F. and Crespi, M., 2017b. 3D modelling of archaeological small finds by a low-cost range camera: methodology and first results. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLII-5/W1, pp. 589–592.