Comparison of RGB-D sensors for 3D reconstruction

José Gomes da Silva Neto
Universidade Federal de Pernambuco
Voxar Labs
Recife, Brazil
jgsn@cin.ufpe.br

Pedro Jorge da Lima Silva
Universidade Federal de Pernambuco
Voxar Labs
Recife, Brazil
pjls2@cin.ufpe.br

Filipe Figueredo
Universidade Federal de Pernambuco
Voxar Labs
Recife, Brazil
ffm@cin.ufpe.br

João Marcelo Xavier Natario Texeiro
Universidade Federal de Pernambuco
Voxar Labs
Recife, Brazil
jmxnt@cin.ufpe.br

Veronica Teichrieb
Universidade Federal de Pernambuco
Voxar Labs
Recife, Brazil
vt@cin.ufpe.br

***Abstract*—The use of low-cost RGB-D sensors became popular with the release of Microsoft Kinect V1, in 2010. From that time on, different companies such as ASUS, Orbbec, and Intel have developed their solutions for capturing 3D data. This work proposes an analysis of how well 10 RGB-D sensors perform while applied to a 3D reconstruction task. The comparison was possible by using COMAU's e.Do robotic Arm for ensuring all sensors performed the same set of 3 different paths while capturing RGB and Depth data from the test environment. A customized version of RTAB-map, an open-source solution for real-time appearance-based mapping, was developed so that all ten sensors were supported by the same tool for capturing data, since RTAB-map's original data recorder module did not support Astra Pro device and Realsense devices were not included in the tool. All evaluations were performed in a controlled indoor scenario, built to address different characteristics on the same scene: highly texturized regions, poorly texturized regions, close and far objects. Almost 20GB of data was collected and is available as a public dataset for future comparison of 3D reconstruction algorithms that receive input from different RGB-D sensors.**

***Keywords*—RGB-D sensor comparison, RGBD analysis, RGB-D benchmark, 3D reconstruction benchmarking, RTAB-map 3D reconstruction, robotics**

I. Introduction

Three-dimensional (3D) reconstruction is the process of capturing the shape and appearance of real objects. This allows the capture of information regarding specific features of the object, which are hard to infer from a single 2D image, such as volume and object's relative position to others in the scene, opening possibilities for numerous applications. A critical area of application of 3D reconstruction is medicine, e.g., allowing 3D tomographic reconstruction of coronary arteries [1]. The utilization of 3D reconstruction in the medical area led to vast improvements, especially in the diagnosis of fetal anomalies [2]. Another area that benefited from 3D reconstruction is architecture, making it possible to study old structures as "The Sarno Baths in Pompeii" [3].

There are many ways of acquiring 3D information from a scene. A direct one is through the use of depth sensors with Time of Flight or Structured Light technology, or RGB sensors, using Stereo Vision technology. The complexity of a color sensor system is mainly based on the number of wavelength bands, or signal channels, it uses to resolve color. Systems can range from a comparatively simple three-channel colorimeter to a multi-band spectrometer. Color Sensors use RGB filters to detect and perform color measurements of objects placed in front of the sensor. Hence, when an object is placed in front of the sensor, it displays the same color through a LED. The sensor works on eight colors: the primary; green, red and blue, the secondary; magenta, yellow and cyan, as well as black and white. The circuit uses optics and digital electronics to detect the color of the object, [4]. An RGB-D image is simply a combination of an RGB image and its corresponding depth image. A registered depth image is an image whose single channel refers to the distance between the image plane and the corresponding object in the RGB image.

Simultaneous Localization and Mapping (SLAM), a technique capable of mapping an unfamiliar space and identifying the camera's location, may benefit from the use of RGB-D sensors [5], [6]. Typical SLAM applications include the areas of autonomous mobile robotics and computer vision [7]. With the diversity of available SLAM approaches, determining which one to use concerning a specific platform and application is a difficult task, mostly because of the absence of comparative analysis between them. SLAM approaches are generally visual-based or LIDAR-based only and are benchmarked often on datasets having only a camera or a LIDAR, but not both, making it difficult to have a meaningful comparison between them. The Robotic Operating System (ROS), introduced in 2008, contributes significantly to standardize sensor data format, thus improving interoperability between robot platforms and making it possible to compare SLAM approaches. But still, visual SLAM approaches integrated in

ROS are not often tested on autonomous robots: only SLAM by teleoperation or by a human moving the sensor, this avoids proper Transformer Library (TF) [8], handling to transform the outputs according to the robot base frame to satisfy ROS coordinate frame convention. It also avoids the need to have map outputs (e.g., 2D or 3D occupancy grid) compatible with the navigation algorithm to plan a path and avoid obstacles.

Some of the practical requirements outlined before, such as interoperability between sensors or tool which can use both sensors, are not always entirely addressed by SLAM approaches. To solve that problem, the Real-Time Appearance-Based Mapping (RTAB-Map), an RGB-D, Stereo and LIDAR Graph-Based SLAM approach based on an incremental appearance-based loop closure detector was developed. RTAB-Map being a loop-closure approach with memory management as its core, is independent of the odometry approach used, meaning that it can be fed with visual odometry, LIDAR odometry, or even just wheel odometry. This means that RTAB-Map can be used to implement either a visual SLAM approach, a LIDAR SLAM approach, or a mix of both, making it possible to compare different sensor configurations on a real robot.

In this work we propose an analysis on how well 10 RGB-D sensors perform while applied to a 3D reconstruction task, providing to community a base to chose the best sensor that fits to requirements of theirs application. The sensors chosen are: Kinect V1; Kinect V2[1]; Asus Xtion PRO live[2]; Orbbec Astra[3]; Orbbec Astra Pro[4]; Realsense ZR300[5]; Realsense D435[6]; Realsense R200[7]; Realsense SR300[8]; Realsense F200[9]. These 10 sensors were chosen considering that they are the most popular among the articles surveyed in the bibliographic review, even if they are no longer commercialized as Kinect V1 and V2, or are new and low-cost technologies when compared to millimeter-precision sensors. To enable the comparison, we used the COMAU® e.Do[10], a 6-DOF (Degrees of Freedom) robotic arm to hold all 10 sensors, one by one, and perform 3 different circular paths for each of them: a simple 360° path, a wavy 360° path and a five-point star-shaped path. All sensor data were captured with 30Hz frequency, as well the pose of the robotic arm.

II. Related Works

This section starts by detailing how the search for works that deal with RGB-D sensors comparison was performed. In sequence, the most prominent material is listed and their strong points highlighted. At last, works related to the use of such sensors are also described, mainly the ones related to 3D Reconstruction, SLAM, and robotics.

A. Research Approach

The search in the state of the art was performed in Google Scholar and began with articles that compare RGB-D sensors. The main keywords used were: “RGB-D sensor comparison”, “RGBD analysis”, “RGB-D benchmark”. In the chosen articles, we evaluated the RGB-D sensors and the main parameters compared.

The second phase of the search focused on articles related to 3D reconstruction using RGB-D sensors. The main keywords used were: “3D reconstruction benchmarking”, “dense 3D reconstruction”, “3D reconstruction comparison”. We evaluated the parameters used to compare the sensors, the target scenario, and the sensors used.

The third phase of the search focused on articles regarding SLAM. The main keyword used was: “SLAM Comparison”. We evaluated the methods, scenarios, and sensors used on each chosen work.

The final phase focused on RTAB-map applied to robotic applications. The main keywords used were: “RTAB-map 3D reconstruction”, “RTAB-map robotic”. We evaluated the methods, scenarios, and sensors used on each chosen work as well.

In total, 56 different papers were analyzed, and the main results are be further described in the following subsections.

To have a feeling on how relevant the choice of the 10 RGB-D sensors used in this work was, we have analyzed all publications from the Symposium on Virtual and Augmented Reality (SVR) [11], from 1997 to 2019, This comprised about 813 papers (including full and short ones). We were able to find 100 works with references to Microsoft Kinect (V1 or V2), four works with references to Asus Xtion, five works referring to Intel Realsense devices, and a single work mentioning Structure.io. The first work to mention an RGB-D device in the conference dates from 2011, one year after Kinect’s release. Our comparison comprises the majority of the aforementioned sensors, including two Orbbec Astra devices.

B. RGB-D Sensors Comparison

Many works regarding RGB-D sensors comparison presents focus on a specific area and compare a limited amount of sensors. The work from Guidi *et al.* [9] presents the highest number of different sensors and different types of technology. In this paper one sensor based on Time of Flight technology was used (Kinect V2), three sensors using Structured Light with the Primesense sensor (Structure.io Sensor by Occipital, Xtion PRO by ASUS, Kinect V1 by Microsoft) and an F200 sensor by Intel/Creative also with Structured Light but with Realsense pattern projection technology. All sensors were tested in an indoor scenario to compare the reconstruction of a glass board and prove the capability of the sensors

[1]https://bit.ly/kinectofficial
[2]https://bit.ly/xtionSpecs
[3]https://bit.ly/astraSpecs
[4]https://bit.ly/astraSpecs
[5]https://bit.ly/zr300specs
[6]https://bit.ly/d435specs
[7]https://bit.ly/r200specs
[8]https://bit.ly/sr300specs
[9]https://bit.ly/f200specs
[10]https://bit.ly/edoSite

[11]The event was named SVR only in 2001. Before this date, the event was known as WRV (Workshop of Virtual Reality), happening in 1997, 1999 and 2000.

for gesture tracking, arguing that the smaller the uncertainty error obtained, the better the sensor would be for this task. To compare all sensors, the authors used systematic errors (associated to the concept of accuracy) and random errors (associated with the concept of precision) and conclude that all sensors have small uncertainties and can be used for gesture tracking and 3D reconstruction.

The work from Gesto Diaz *et al.* [10] compares an Asus Xtion and a Microsoft Kinect V2 in an indoor scenario for an object recognition task. The scenario was composed of many objects made of basic primitives and objects with different shapes, sizes and textures such as toys, cups and even a coffee box. To generate the ground truth to compare the sensors, a Hexagon Metrology Absolute Arm 7325SI was used. The comparison parameters were the accuracy, based on radiometric completeness, spatial completeness and environment statical analysis, and object recognition analysis, based on object recognition rate and 6-DOF pose estimation. After the results, the authors conclude that Kinect has better results than ASUS Xtion, which had more fitted points but presented problems during object recognition. Another critical point is that Kinect can work in outdoor environments, and this is not possible to Xtion due to its technology based on structured light.

C. 3D Reconstruction and SLAM Works

For 3D reconstruction, SLAM algorithms and trajectories comparison, many studies provide large datasets composed of different scenarios and respective ground truths. An RGB-D benchmark captured by four sensors is presented in reasearch of Song *et al.* [11]. This study aims to construct a large dataset for object reconstruction and recognition tasks using four RGB-D images produced by an Intel Realsense, ASUS Xtion Microsoft Kinect V1 and V2. This study presented many metrics for comparing algorithms but did not provide a more in-depth comparison regarding technologies used to scan the environments and objects.

In the work from Wang *et al.* [12], three different sensors are used to measure the distance between camera and fruits on trees to estimate the fruit size. During a preliminary experiment to determine the Root Mean Square Error of each sensor, the ZED stereo camera presented an inferior result to the other two cameras, and so the test with fruits was discarded. LIDAR had the smallest error, with Kinect V2 in second place. However, in the experiment to estimate the size of the fruits neither of the two depth cameras could be used to measure the fruits directly due to the small resolution.

Another benchmark is provided by Handa *et al.* [13]. This study uses a handheld camera to capture images of a living room and an office scene to compare different SLAM algorithms. Although the research did not focus on sensors, it provided essential metrics to compare trajectories such as Absolute Trajectory Error (ATE), which can be used to compare sensors since the SLAM algorithms stay the same.

D. RTAB-map for Robotics

As mentioned before, RTAB-map is an RGB-D, Stereo and LIDAR Graph-Based SLAM approach based on an incremental appearance-based loop closure detector [14]. To determine how likely a new image comes from a previous location or a new location, the loop closure detector uses a bag-of-words approach. When a loop closure hypothesis is accepted, a new constraint is added to the map's graph, and then a graph optimizer minimizes the errors in the map. A memory management approach is used to limit the number of locations used for loop closure detection and graph optimization so that real-time constraints on large-scale environments are always respected. RTAB-Map can be used along with different RGB-D sensors, a stereo camera or a 3D LIDAR for 6-DOF mapping, or on a robot equipped with a laser rangefinder for 3-DOF mapping.

The reason RTAB-map was chosen instead of other well-known solutions, such as ORBSLAM2 [15], is that according to Labbé *et al.* [14], RTAB-map supports online output of dense point cloud, while ORBSLAM2 does not have this feature.

III. Hardware for the Experiments

As stated before, there are many ways of capturing 3D data from a real scene. This section does not aim to be exhaustive in explaining all possible methods for doing that. Instead, it focuses on the technologies behind the ten different image sensors used in the comparison made and also on the robotic arm specifications that had an impact on data acquisition.

A. Technologies for Depth Measurement

There are now many techniques for measuring objects and depth. For this purpose, simpler sensors, such as ultrasonic sensors that use sound waves, can be used. More robust sensors that use images and light patterns or even lasers, estimating the depth based on the time the light travels over a given distance are generally used for more complex scenes and tasks such as SLAM or object recognition.

Figure 1 presents an overview of different technologies for the acquisition of depth information. We decided to focus on technologies derived from the "Light Waves" classification, while "Microwaves" and "Ultrasonic Waves" based technologies are out of the scope of this work since we are dealing with visual information.

In the following sections, each of the technologies covered are be briefly described.

1) Active Stereo Sensors: Stereo cameras are based on a technique that uses spatial displacement between a pair of images to triangulate and calculate the real distance between objects and camera. Many systems use feature extraction algorithms and they can present some issues while trying to identify key points in scenarios with poor textures. According to Jang *et al.* [17], to avoid these problems, active stereo cameras use light patterns such as random dots, sinusoidal waves, or stripes to improve the search of correspondent key points.

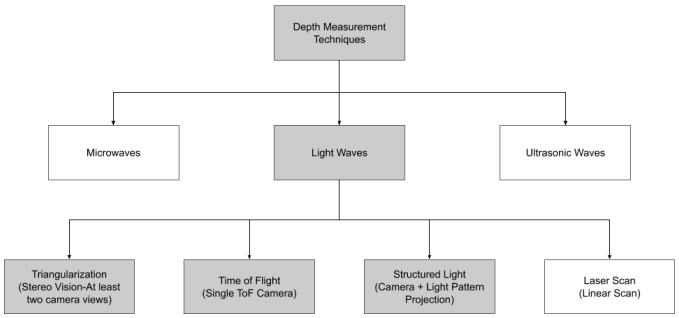

Fig. 1. Different technologies for the acquisition of depth maps. Modified from Escalera [16].

2) Structured Light Sensors: Cameras that use structured light can be split into different classifications. The first one, called *temporally coded patterns*, uses more than one pattern to create the image and differentiate the visual elements; however, this approach requires synchronization between the patterns projector and the camera. Differently, *spatially coded patterns* use spatial neighborhood similarities and do not require multiple patterns. However, this technique can present problems in edges or if there is occlusion of the object in relation to the projector.

3) Time of Flight Camera: Time of Flight (ToF) technology is commonly used in LIDARs. According to Horaud *et al.* [18], ToF technology can use emitted pulses of laser beams and measure the light round-trip time or project a continuous-wave and measure its phase shift, obtaining the time of flight indirectly. Both approaches provide precise distance information.

Unlike LIDARs, RGB-D sensors do not use a single shot pulse but instead emits a cone of shots, obtaining information regarding all image pixels. This technology allows the use in outdoor scenarios and under adverse conditions with less impact in distance information quality. Scenarios with poorly reflective objects may impact negatively measurements made with this technology.

B. Technical Specifications of Selected RGB-D Sensors

The technical specifications regarding all 10 RGB-D sensors selected for this work are presented in Table I. All information was obtained from the websites of the sensors' manufactures. Some discontinued models, such as the Intel Realsense F200, have less information available.

It is important to note that between the ten cameras used to this research, only Kinect V2 uses the Time of Flight technology. Six Cameras uses Structured Light (Kinect V1, ASUS Xtion, Astra, Astra Pro, SR300, and F200), and three uses Active Stereo technology (ZR300, D435, and R200).

Two of them work in short-range, R200, and SR300. The D435 has the longest range working as well to short and too long distances. The Astras cameras work well in a middle range until 8m, and the Kinects are the only cameras that need external supply making it difficult to use in mobile platforms or scenarios where the power supply is limited.

Most sensors have a 640x480 depth image resolution and a 30fps capture rate, with the exception of the D435 which can capture up to 90fps. The RGB sensors of most cameras are either HD or Full HD with a capture rate of 30fps.

C. COMAU's e.Do Specification

The robotic arm used on this project has the goal of holding the RGB-D sensors, using its gripper, while performing three predetermined paths. Such motion paths are more detailed in the next section. The e.Do[13] is a 6-DOF robotic arm mainly used on educational applications, but it has the advantage of being open-source and low cost in comparison to other robotic arms with a similar accuracy motion. e.Do's first three axes (also called joints) #1, #2, and #3 have a maximum velocity of 38°/second. The other three axes, #4, #5, and #6, have a maximum velocity of 58°/second. The gripper is considered as joint #7.

The main reason that made e.Do essential for this research is that it helps to guarantee all RGB-D sensors perform the same trajectories while capturing data. We chose to use the e.Do arm with only 25% of its maximum velocity, to make sure it would be slow enough to not interfere in the 3D reconstruction (by increasing the baseline, since a faster movement necessarily implies in consecutive frames with more differences), and thus allowing the trajectory data provided as ground truth for the trajectory. It is also important to highlight some of e.Do movement limitations, as pointed out in Table II. For example, the base joint (joint #1) is not able to perform a complete 360° spin, since it goes from -178.90° to 178.90°. Fortunately, this fact does not harm the capture process using the arm.

IV. Methodology

This section will emphasize how the evaluation of the sensors was designed together with some discussion regarding the 3D reconstruction comparison, the result of the different captures performed. Differently from the work of [19], which directly compares the output of the sensors, in the proposed work, we will compare the results after processing sensor data using the same reconstruction algorithm. This will help us perceive how good a specific sensor performs when 3D reconstruction tasks are targeted.

A. Capture Environment

The work starts with the definition of the scene to be captured. In times like this, where access to laboratory facilities and other common places are limited due to the COVID-19 pandemic, such a controlled environment was adapted inside a 2.85x3.40m child bedroom. The e.Do robotic arm was placed in the center of the bedroom, while it was filled with many objects so that there were different texture information densities on its surroundings. Figure 2 illustrates the capture environment from four distinct points of view (pictures were

[12]Not informed by sensor's data-sheet
[13]https://bit.ly/edoBrochure

TABLE I
TECHNICAL SPECIFICATIONS FOR EACH OF THE 10 RGB-D SENSORS SELECTED.

Sensor	Kinect V1	Kinect V2	Xtion	Astra
Type	Structured Light	Time of Flight	Structured Light	Structured Light
Min∼Max Range	0.8m ∼4m	0.5m ∼4.5m	0.8m ∼3.5m	0.6m ∼8m
Depth FoV(HxVxD)	57°x43°	70°x60°	58°x45°x70°	60°x49.5°x 73°
RGB FoV(HxVxD)	57°x43°	84.1°x53.8	58°x45°x70°	60°x49.5°x73°
Depth Resolution@fps	640x480@30fps	512x424@30fps	640x480@30fps	640x480@30fps
RGB Resolution@fps	640x480@30fps	1920x1080@30fps	1280x1024@30fps	640x480@30fps
Size	305x76.2x63.5mm	66x249x67mm	180x35x50mm	165x30x40mm
Power Supply	USB 2.0 and external	USB 3.0 and external	USB 2.0	USB 2.0
OS	Windows/ROS	Windows/ROS	Android/Linux/Windows	Android/Linux/Windows
SDK	Kinect SDK V1	Kinect SDK V2	OpenNI	Astra SDK or OpenNI
Power Consuption	12W	15W	<2.5W	<2.4 W

Sensor	Astra Pro	ZR300	D435
Type	Structured Light	Active Stereoscopy	Active Stereoscopy
Min∼Max Range	0.6m ∼8m	0.55m ∼2.8m	0.105 m ∼10m
Depth FoV(HxVxD)	60°x49.5°x73°	59±5%x46±5%x70±5%	87°±3 °× 58°±1°× 95°±3 °
RGB FoV(HxVxD)	60°x49.5°x73°	68±2%x41.5±2%x75±4%	69.4°× 42.5°× 77°(± 3°)
Depth Resolution@fps	640x480@30fps	628x468@30fps	1280×720. Up to 90 fps.
RGB Resolution@fps	1280x720@30fps	1920x1080@30fps	1920 × 1080 @30fps
Size	165x30x40mm	Not found	90x25x25mm
Power Supply	USB 2.0	USB 3.0	USBC
OS	Android/Linux/Windows	Windows/Linux	Windows/Linux
SDK	Astra SDK or OpenNI	Realsense SDK or LibRealsense	Intel Realsense SDK 2.0
Power Consuption	<2.4 W	<2W	3.5W

Sensor	R200	SR300	F200
Type	Active Stereoscopy	Structured Light	Structured Light
Min∼Max Range	0.4m ∼2.8m	0.2m ∼1.5m	[12]
Depth FoV(HxVxD)	59°± 5°× 46°± 5°× 70°± 4.5°	71.5°± 2°× 55°± 2°× 88°± 3°	73°x59°x90°
RGB FoV(HxVxD)	70°± 2°× 43°± 2°× 77°± 4°	68°± 2°× 41.5°± 2°× 75.2°± 4°	[12]
Depth Resolution@fps	640 x 480 @30fps	640x480 @30fps	640x480@30fps
RGB Resolution@fps	1920 x 1080 @30fps	1920x1080 @30fps	[12]
Size	101.56x9.55x3.8mm	110x12.6x 4.1mm	110x12.5x3.75mm
Power Supply	USB 3.0	USB 3.0	USB 3.0
OS	Windows/Linux	Windows/Linux	[12]
SDK	LibRealsense	LibRealsense	[12]
Power Consuption	1.7 W	1.8W	[12]

TABLE II
E.DO JOINTS MINIMUM AND MAXIMUM ANGULAR POSITIONS. VALUES EXPRESSED IN ANGLES, EXCEPT FOR THE GRIPPER (JOINT #7), EXPRESSED IN MM.

Sensor	Angular Position Variation
Joint #1	-178.90°to 178.90°
Joint #2	-98.92°to 98.92°
Joint #3	-98.92°to 98.92°
Joint #4	-178.91°to 178.91°
Joint #5	-103.41°to 103.41°
Joint #6	-178.91°to 178.91°
Joint #7	0 to 80.53mm

taken from the upper corners of the bedroom). All recordings were performed late at night when no external light was noticed from the closed window. The bedroom's light was turned on all the time, guaranteeing the environment's isonomy for all sensors. This is very important to evaluate how well the reconstruction algorithm performs when dealing with data with varying textures and shapes captured using different sensors.

B. Data Capture

The data capture is divided into two phases. The first one is related to the three trajectory paths proposed so that each sensor would perform the same movements. The three paths are shown in Figure 3. The reason why the three paths start and stop at almost the same position is that this helps to evaluate the performed trajectories, meaning that it is easier to identify if the camera path captured starts and ends at the same point in space. Each path has different but noticeable characteristics. While the first path comprises a simple 360° spin around the robotic arm (just by rotating joint #1), the second path represents a wavy movement, performing the same spin as before but now varying the Z coordinate in an up-down periodic movement. At last, the third path focuses on changing the distance to objects and how this would impact the reconstruction. While performing the 360° spin, the robotic arm contracts itself, moving the gripper closer and away from

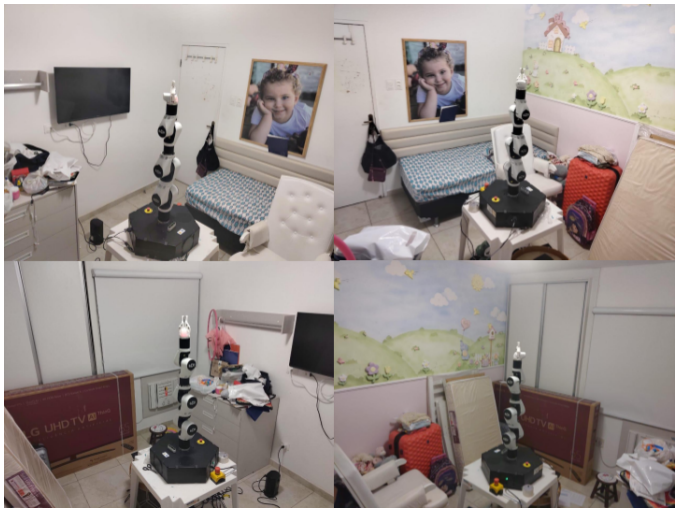

Fig. 2. Four different views of the capture environment.

the bedroom's center. All 3 paths were executed with the same speed and acceleration in order to exclude these factors from the reconstruction, leaving only the path to influence the capture of images.

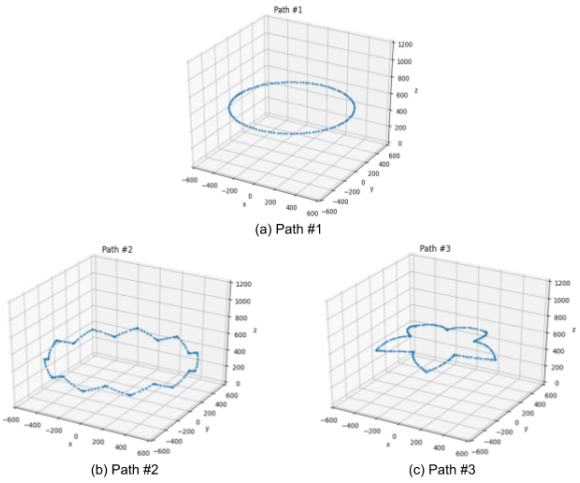

Fig. 3. All paths used to capture images. Scale in millimeters.

Figures 4, 5 and 6 show sequences of images captured by the Astra Pro device. They illustrate the sensor point of view during the paths #1, #2 and #3 respectively.

The second phase of the data capture is related to the setup of RTAB-map and its challenges. To collect the data from the sensors, we used RTAB-Map's data recorder module. Unfortunately, some of the sensors we wanted to test, for example, Astra Pro, could not be used to record data through the data recorder module available in RTAB-Map's online repository. To overcome this problem, it was necessary to recompile RTAB-Map and add some of the missing support for the devices in the data recorder module. It is important to say that RTAB-Map executable already supports most of the sensors, but the 3D reconstruction is made during capture phase with data recorder module. Since we wanted to only perform the capture and later process the data, the data recorder module was necessary. The modified version of RTAB-Map and its data recorder module is available at the project's website[14].

C. 3D Reconstruction Comparison

The data comparison is based on two different analyses. The first one is a quantitative analysis based on comparing the result point clouds. Based on that information, we evaluate the number of points obtained. This, together with a visual analysis of the generated point clouds, provides an indication of which sensor captures better the chosen environment. The quantitative analysis also includes a comparison between the ground truth path executed by the robotic arm and the camera path previously extracted from RTAB-Map. This is better discussed in the next section.

To compare the path obtained after the RTAB-map processing, the error between the trajectories was calculated. The Euclidean distance for every pair of point is given by Equation (1), where $\|r - p_r\|$ denotes the Euclidean norm, p_r is the corresponding point in set P that attains the shortest distance to a given point r in set R.

$$dist(R, P) = \sum_{r \in R} \|r - p_r\|, \; p_r = \arg\min_{p \in P} \|p_r - r\| \quad (1)$$

The second phase of comparison is qualitative analysis. Initially, the three paths are visually compared to the ground truth trajectory provided by e.Do. After that, we compare the 3D reconstruction of the bedroom, using a top view image of the reconstruction based on through the generated point cloud. In this phase, we analyze the accuracy of object reconstruction, how precise the room representation is, and which works better for close and far objects.

During the research of state of the art, a few works already show some of the sensors perform better than the other. After all the comparison made by us, we also look to explain more details, why of this results.

V. Results and Discussion

This section reports the results obtained with the experiments and analyzes them, both quantitatively and qualitatively.

A. Quantitative Results

The initial analysis regards the camera trajectory. To each path, we compare the sensor information and the ground truth provided by e.Do. The error is calculated based on the trajectory points sampled every 0.5 seconds from e.Do. A total of 133, 210, and 188 points were sampled for trajectories #1, #2 and #3, respectively. This also means that trajectories #1, #2 and #3 took 66.5, 105, and 94 seconds to be performed. Path #2 takes longer to be completed because there are more control points and "move joint" commands sent to the robot. Since the robot accelerates when leaving its current position

[14]https://rgbd10.github.io/svr2020cag/

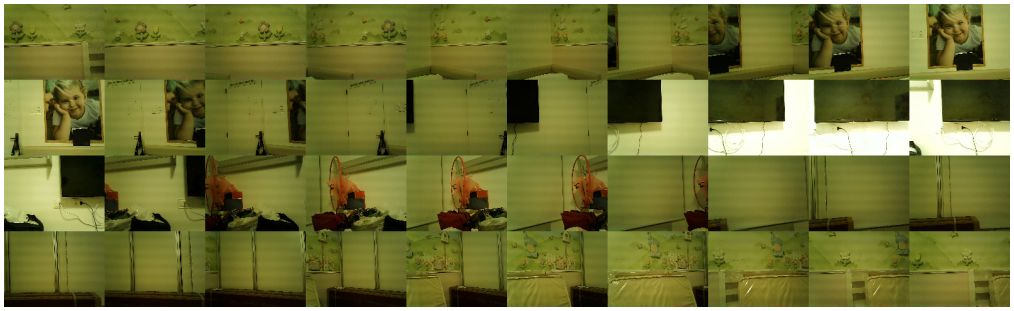

Fig. 4. Sequence of images captured by Astra Pro camera during path #1.

Fig. 5. Sequence of images captured by Astra Pro camera during path #2.

Fig. 6. Sequence of images captured by Astra Pro camera during path #3.

TABLE III
SIZE OF .DB FILES (IN MBYTES) BEFORE AND AFTER BEING PROCESSED BY RTAB-MAP.

	Before processed by RTAB-Map			**After processed by RTAB-Map**		
Sensors	**Path #1**	**Path #2**	**Path #3**	**Path #1**	**Path #2**	**Path #3**
Astra	327	540	492	696	1157	1280
Astra pro	166	275	245	295	593	507
D435	351	550	500	505	311	930
F200	603	1075	693	215	507	454
Kinect V1	309	523	462	548	1106	1096
Kinect V2	697	1085	988	1178	2099	1987
R200	734	1331	843	248	353	398
SR300	694	1219	843	297	596	381
Xtion	214	379	329	551	1126	1106
ZR300	798	1188	1198	727	867	1751

and then deaccelerates when arriving on the destination, the increase in the number of commands is directly proportional to the total time to perform the movement.

After converting the reference points from the robot to meters and changing them to the camera coordinate system of each sensor, the error was calculated.

Figures 7, 8 and 9, show the mean error for each sensor, in path #1, #2 and #3, respectively. Analyzing the three charts, it is possible to perceive that most of the Intel Realsense sensors show the highest error values. For path #1, all Realsense

sensors performed worse, with the exception of D435, which is a newer version in comparison to the other ones.

Taking into account the three errors computed for each path, as shown in Table IV, the Astra Pro device presented the best result, followed by Xtion, Astra, Kinect V2 and Kinect.

Fig. 7. Mean error for path #1. Scale in centimeters.

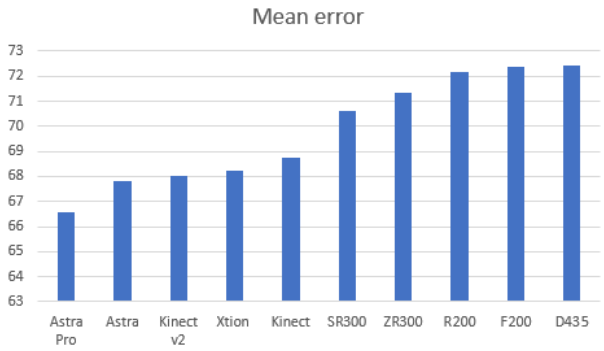

Fig. 8. Mean error for path #2. Scale in centimeters.

Fig. 9. Mean error for path #3. Scale in centimeters.

As result of RTAB-map processing, we obtained 30 point clouds. Table V show the amount of points, the average, variance and standard deviation for each path obtained by each sensor.

B. Qualitative Results

Regarding path #1, all sensors were capable of performing a full 360 reconstruction of the bedroom, except the older Realsense models (F200, R200, SR300, and ZR300), as shown in Figure 10. The camera path retrieved was coherent in all full 360° reconstruction cases. From the sensors that did not manage to complete the full 360° reconstructions, the ZR300 model achieved about 3/4 of it.

Regarding path #2, all sensors were capable of performing a full 360° reconstruction of the bedroom, except this time, no Realsense model was able to do it. It seems both D435 and ZR300 had trouble and lost the reconstruction in the area close to the TV, which is the most challenging one with less features a large "dead reflective area" from the TV image, as shown in Figure 8.

Regarding path #3, all sensors were capable of performing a full 360 reconstruction of the bedroom, except the Realsense models F200, R200 and SR300, as shown in Figure 9. These tree sensors seemed to have problems while acquiring depth information of the first corner of the path. This behavior was expected for the F200 model, which works as a front camera (aiming users face), but not with R200 (rear camera) and the SR300, these last two with 0.4-2.8m and 0.2-1.5m range, respectively.

TABLE IV
MEAN ERROR TO EACH POINT OF GROUND TRUTH PATH. INFORMATION IN CENTIMETERS.

Sensor	Path #1	Path #2	Path #3	Error Sum
Astra	56.80	67.79	22.89	147.48
Astra pro	57.09	66.55	22.61	146.25
D435	57.04	72.43	22.19	151.66
F200	60.38	72.39	30.83	163.59
Kinect v1	58.42	68.77	23.19	150.38
Kinect v2	56.12	68.03	24.03	148.18
R200	59.48	72.16	31.24	162.88
SR300	60.91	70.61	33.40	164.91
Xtion	55.91	68.22	23.32	147.45
ZR300	59.49	71.34	22.59	153.41
Mean	57.76	69.69	23.26	151.02
Max. value	60.91	72.43	33.40	164.91
Min. value	55.91	66.55	22.19	146.25
Variance	3.28	4.83	18.94	54.07
Std. deviation	1.81	2.20	4.35	7.35

TABLE V
QUANTITY OF POINTS REGISTERED IN THE POINT CLOUD.

Sensor	Path #1	Path #2	Path #3
Astra	473872	411599	484051
Astra pro	**703207**	**788659**	742904
D435	573488	305918	665815
F200	74088	168736	127451
Kinect v1	572270	506604	483317
Kinect v2	694170	644879	920100
R200	**71932**	**83845**	108320
SR300	117054	121204	**87979**
Xtion	539903	504957	556562
ZR300	542159	242593	**1101094**
Mean	541031	358758,5	520306,5
Max. value	703207	788659	1101094
Min. value	71932	83845	87979
Variance	6.27E+10	5.48E+10	1.20E+11
Std. Deviation	2.50E+05	2.34E+05	3.47E+05

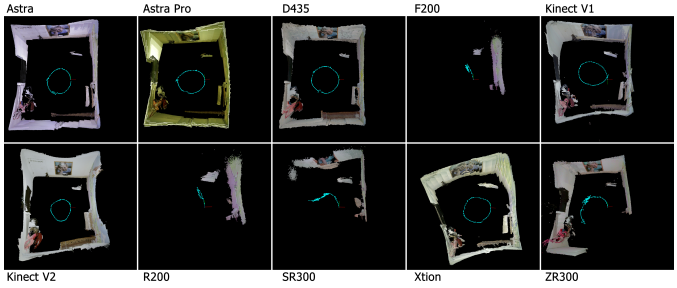

Fig. 10. Resultant path #1 and point cloud reconstruction for each device.

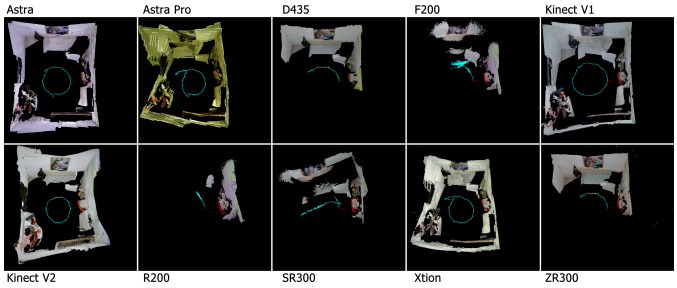

Fig. 11. Resultant path #2 and point cloud reconstruction for each device.

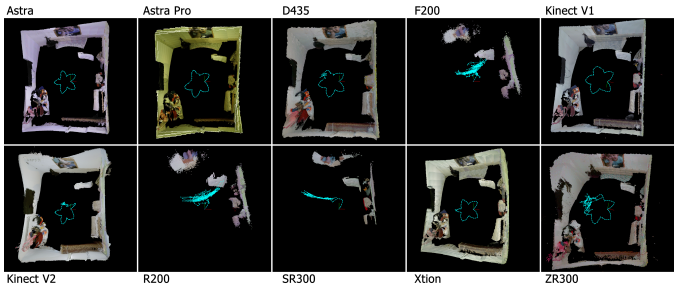

Fig. 12. Resultant path #3 and point cloud reconstruction for each device.

C. Discussion

This section describes what was learned from the reconstruction analysis performed, together with the pose/path extracted from the sensor movement.

1) Reconstruction Analysis: After analyzing the quantitative results, it is possible to verify that for path #1 five sensors presented a point cloud with more points than the mean: Astra Pro, Realsense D435, Kinects V1 and V2 and ZR300. Astra and Xtion presented less points than the mean but Realsense F200, R200, SR300 presented significantly less points than the average. However, it is important to note that ZR300 presented an incomplete reconstruction even presenting more points than the mean and Astra and Xtion presented a complete reconstruction of the scenario even with an amount of points under the average. F200, R200 and SR300 presented an incomplete reconstruction, what was already expected due to the small amount of points. In this test, Astra Pro obtained the point cloud with most point while R200 with lesser points.

Regarding path #2, Astra Pro, Astra, Kinects V1 and V2 and Xtion presented clouds with more points than the average but Astra Pro and Xtion presented a reconstruction with a lower quality when compared to Astra and both Kinects. D435 presented a point cloud with less points than the average but not too much, however presented an incomplete reconstruction. SR300, ZR300, F200 and R200 presented a number of points far below the average and an incomplete reconstruction with R200 presenting the worst result. D435 and ZR300, even with very different amounts of points, $305,918$ and $242,593$, respectively, presented a very similar reconstruction. In this test, Astra Pro obtained the point cloud with the most points and R200 obtained the cloud with the least points.

In results presented for path #3, Astra Pro, D435, Kinect V2, Xtion and ZR300 obtained more points than the mean. Between them, only ZR300 presented a reconstruction with problems mainly in the upper left corner (this is problem is easy to visualize if one looks at the generated camera path). In this path, Kinect V1 and Astra generated a point cloud with less points than the mean but presented a reconstruction closer to reality than ZR300. F200, R200 and SR300 generated point clouds with less points than the mean and presented incomplete and very similar reconstruction. In this test, ZR300 obtained the point cloud with the most points while SR300 obtained the point cloud with the least points.

Analyzing the results, R200 e SR300 were always among the worst reconstructions. The point clouds generated with these cameras always contain less points than the mean and are out of the standard deviation margin. One possible reason for that may be the short range characteristic of those cameras. Despite Realsense sensor specifications on the manufacturer website, some of them do not work adequately with 3D reconstruction in the general range specified. For instance, this page [15] states that for 3D scanning purposes, the F200 sensor only works in the range between $27 - 54$cm.

The Astra Pro camera generated point clouds with more points than the mean in all three paths and in path #2 it generated a point cloud with more points than the standard deviation margin. Also, Kinect V2 performed above the upper standard deviation margin in path #2 and #3. ZR300 performed above the upper standard deviation margin in path #3.

2) Pose/Path Analysis: Analyzing the path results from the reconstruction, it is possible to verify that for path #1 five cameras presented a path with a mean error lower than the mean of the errors: Astra Pro, Astra, Realsense D435, Kinect V2, and Xtion. Kinect V1 presented a mean error higher than the average but below the upper limit of standard deviation and the path generated is compatible with the ground truth. Realsense ZR300 and R200 also presented a mean error below the upper limit of standard deviation but differently from Kinect V1, they did not generate a complete path after reconstruction. F200 and SR300 presented a mean error higher than the upper limit of standard deviation and also presented a failure on path reconstruction with an incomplete path. In this test, Xtion obtained the reconstructed path with the lowest

[15] https://bit.ly/f200Specs

mean error and SR300 obtained the reconstructed path with the highest mean error.

Analyzing the results of path #2, Astra Pro, Astra, Kinects V1 and V2 and Xtion presented a mean error lower than the mean of the errors. However, Astra Pro, Astra and Xtion presented some problems in their reconstructed path such as drifts. Both Kinects presented a reconstructed path closer to ground truth. R200, ZR300, F200, SR300 and D435 presented a path with mean error higher than the mean of the errors and incomplete path reconstruction. In this test, Astra Pro obtained the reconstructed path with the lowest mean error while R200 obtained the reconstructed path with the highest mean error.

In the results presented for path #3, Astra Pro, Astra, D435, Kinect V1 and ZR300 performed with more points than the mean. However, ZR300 and D435 presented some problems in their reconstructed path such as drifts. Kinect V2 and Xtion presented a mean error higher than the mean of the errors but also presented a reconstructed path closer to ground truth. R200, F200 and SR300 presented a path with mean error higher than the mean of the errors and incomplete path reconstruction. In this test, D435 obtained the reconstructed path with the lowest mean error while SR300 obtained the reconstructed path with the highest mean error.

VI. Conclusion

This work performed a comparison of 10 different RGB-D sensors regarding their ability to generate data for 3D reconstruction applications. To enable a fair comparison among all sensors, a customized version of the RTAB-Map tool was recompiled. This customized version includes support for Astra Pro, which was not previously supported by official RTAB-map releases. Regarding the comparison of the sensors itself, to the best of our knowledge, the proposed work is the one that compares the most number of RGB-D sensors, followed by [9], comparing 5 RGB-D sensors.

According to the previous sections, most sensors performed well and were able to perform a complete 360 degrees reconstruction of the scene. The worst results pertained to the Realsense sensors, mainly the older ones, probably due to their range and technology behind its structured light pattern. Considering the path errors for all three paths, Astra Pro, Xtion, Astra, Kinect V2, and Kinect V1 were the sensors with the best results, in descending order. It is important to notice that despite both Kinect versions showing good results, their use is limited due to the fact these sensors need auxiliary power. Both Astra versions and Xtion, together with D435 (the newest Realsense model amongst the ones tested), are good choices when performing large scale reconstructions using mobile platforms (robots, tablets, etc).

We made available[16] all data regarding the 10 RGB-D captures. This comprises over 19.2GB of data, considering both RGB and Depth streams for all cameras. Besides that, it is also available the generated point clouds, camera poses, ground truth information from the robot path, the recompiled version of RTAB-Map and its data recorder module. Such data collection may be a valuable dataset in the future for testing different 3D reconstruction algorithms, in case one wants to guarantee that it performs well using different RGB-D sensors.

[16]https://rgbd10.github.io/svr2020cag/

References

[1] C. Blondel, R. Vaillant, G. Malandain, and N. Ayache, "3d tomographic reconstruction of coronary arteries using a precomputed 4d motion field," *Physics in Medicine & Biology*, vol. 49, no. 11, p. 2197, 2004.

[2] H. Werner Jr, "3d reconstruction in fetal medicine," *Ultrasound in Medicine & Biology*, vol. 45, pp. S44–S45, 2019.

[3] L. Bernardi, M. S. Busana, V. Centola, C. Marson, and L. Sbrogiò, "The sarno baths, pompeii: architecture development and 3d reconstruction," *Journal of Cultural Heritage*, vol. 40, pp. 247–254, 2019.

[4] A. Kaushik and A. Sharama, "Rgb color sensing technique," *Int. J. Adv. Res. Sci. Eng*.

[5] R. Liu, J. Shen, C. Chen, and J. Yang, "Slam for robotic navigation by fusing rgb-d and inertial data in recurrent and convolutional neural networks," in *2019 IEEE 5th International Conference on Mechatronics System and Robots (ICMSR)*. IEEE, 2019, pp. 1–6.

[6] S. A. Scherer and A. Zell, "Efficient onbard rgbd-slam for autonomous mavs," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 1062–1068.

[7] S. Huang, C. Li, Z. Cai, G. Zhu, L. Yao, and Z. Fan, "Synchronized 2d slam and 3d mapping based on three wheels omni-directional mobile robot," in *2019 IEEE 9th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*. IEEE, 2019, pp. 1177–1181.

[8] T. Foote, "tf: The transform library," in *2013 IEEE Conference on Technologies for Practical Robot Applications (TePRA)*. IEEE, 2013, pp. 1–6.

[9] G. Guidi, S. GONIZZI BARSANTI, L. L. Micoli *et al.*, "3d capturing performances of low-cost range sensors for mass-market applications," in *23rd International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences Congress, ISPRS 2016*. International Society for Photogrammetry and Remote Sensing, 2016, pp. 33–40.

[10] M. Gesto Diaz, F. Tombari, P. Rodriguez-Gonzalvez, and D. Gonzalez-Aguilera, "Analysis and evaluation between the first and the second generation of rgb-d sensors," *IEEE Sensors Journal*, vol. 15, no. 11, pp. 6507–6516, 2015.

[11] S. Song, S. P. Lichtenberg, and J. Xiao, "Sun rgb-d: A rgb-d scene understanding benchmark suite," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 567–576.

[12] Z. Wang, K. B. Walsh, and B. Verma, "On-tree mango fruit size estimation using rgb-d images," *Sensors*, vol. 17, no. 12, p. 2738, 2017.

[13] A. Handa, T. Whelan, J. McDonald, and A. J. Davison, "A benchmark for rgb-d visual odometry, 3d reconstruction and slam," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 1524–1531.

[14] M. Labbé and F. Michaud, "Rtab-map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation," *Journal of Field Robotics*, vol. 36, no. 2, pp. 416–446, 2019.

[15] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[16] S. Escalera, "Human behavior analysis from depth maps," in *International Conference on Articulated Motion and Deformable Objects*. Springer, 2012, pp. 282–292.

[17] W. Jang, C. Je, Y. Seo, and S. W. Lee, "Structured-light stereo: Comparative analysis and integration of structured-light and active stereo for measuring dynamic shape," *Optics and Lasers in Engineering*, vol. 51, no. 11, pp. 1255–1264, 2013.

[18] R. Horaud, M. Hansard, G. Evangelidis, and C. Ménier, "An overview of depth cameras and range scanners based on time-of-flight technologies," *Machine vision and applications*, vol. 27, no. 7, pp. 1005–1020, 2016.

[19] B. Langmann, K. Hartmann, and O. Loffeld, "Depth camera technology comparison and performance evaluation." in *ICPRAM (2)*, 2012, pp. 438–444.