Article

A Benchmark of Popular Indoor 3D Reconstruction Technologies: Comparison of ARCore and RTAB-Map

Ádám Wolf [1,2,3,*] **, Péter Troll** [4,5] **, Stefan Romeder-Finger** [1] **, Andreas Archenti** [4,5] **, Károly Széll** [2,3,6] **and Péter Galambos** [2,3]

1 Baxalta Innovations GmbH, A-1221 Vienna, Austria; stefan.romederfinger@takeda.com
2 Doctoral School of Applied Informatics and Applied Mathematics, Óbuda University, H-1034 Budapest, Hungary; szell.karoly@amk.uni-obuda.hu (K.S.); peter.galambos@irob.uni-obuda.hu (P.G.)
3 Antal Bejczy Center for Intelligent Robotics, Óbuda University, H-1034 Budapest, Hungary
4 Department of Production Engineering, KTH Royal Institute of Technology, SE-114 28 Stockholm, Sweden; ptroll@kth.se (P.T.); archenti@kth.se (A.A.)
5 Department of Sustainable Production Development, KTH Royal Institute of Technology, SE-151 36 Södertälje, Sweden
6 Alba Regia Technical Faculty, Óbuda University, H-8000 Székesfehérvár, Hungary
* Correspondence: adam.wolf@takeda.com

check for updates

Abstract: The fast evolution in computational and sensor technologies brings previously niche solutions to a wider userbase. As such, 3D reconstruction technologies are reaching new use-cases in scientific and everyday areas where they were not present before. Cost-effective and easy-to-use solutions include camera-based 3D scanning techniques, such as photogrammetry. This paper provides an overview of the available solutions and discusses in detail the depth-image based Real-time Appearance-based Mapping (RTAB-Map) technique as well as a smartphone-based solution that utilises ARCore, the Augmented Reality (AR) framework of Google. To qualitatively compare the two 3D reconstruction technologies, a simple length measurement-based method was applied with a purpose-designed reference object. The captured data were then analysed by a processing algorithm. In addition to the experimental results, specific case studies are briefly discussed, evaluating the applicability based on the capabilities of the technologies. As such, the paper presents the use-case of interior surveying in an automated laboratory as well as an example for using the discussed techniques for landmark surveying. The major findings are that point clouds created with these technologies provide a direction- and shape-accurate model, but those contain mesh continuity errors, and the estimated scale factor has a large standard deviation.

Keywords: 3D scanning; 3D metrology; 3D reconstruction; ARCore; RTAB-Map

1. Introduction

Before the focus can be set on two specific widely available technologies, the big picture of three-dimensional (3D) reconstruction approaches has to be presented. As such, 3D reconstruction is one of the most complex forms of optical sensing, in that it is derived through multiple steps from simpler sensing techniques [1]. Fundamentally, optical sensors are a diverse group of measuring devices, the operation of which is based on retrieving information with the help of the visible spectrum of the electromagnetic waves (referred to as light). This is sometimes extended with the infrared and the ultraviolet spectra. In Figure 1, a framework is provided to place imaging technologies in a hierarchical structure. Each step of deriving a method from a simpler one is denoted by the type of augmentation. Single units can either be combined into vectors or arrays, or they can be given additional degrees-of-freedom by movement.

To get 3D spatial information, two paths are discussed. The stem of the branch on the right is the so-called time-of-flight (ToF) sensor or ranger, which provides 1D information based on measuring the time between emitting and receiving a light signal. To take this type of measurement to two dimensions, the single sensor can be mounted on a rotating platform, and—similarly to a radar—a planar space can be surveyed in a sweeping fashion. This method is called light detection and ranging (LiDAR). 3D coverage, on the other hand, can be achieved by giving the single sensor another degree of freedom (DoF), resulting in a so-called 3D LiDAR. A single ToF unit can also be augmented into a 3D imaging device by combining many of them into a 2D matrix [2]. In this case, the whole scene has to be illuminated with the specially-modulated light signal, which, after being reflected from the objects in the scene, is focused on the sensor matrix by a lens. As an end-result, a depth-image is created, which means that each pixel of the resulting image has a depth value derived from the range measurement of the corresponding ToF unit.

In the hierarchy represented in Figure 1, as the most basic form of optical sensing, a single photosensor, such as a phototransistor, is considered. A single photosensor can be categorised as a spatially null-dimensional (0D) source of information, in that it measures in a single point. However, when multiple photosensors are arranged in a linear array, the resulting derived sensor qualifies as spatially one-dimensional (1D), since it can provide information along one single direction. Analogously, if a 2D matrix is created, the provided information becomes spatially two-dimensional (2D). If an image is projected onto a 2D photosensor matrix, the sensor can be considered as a digital camera. In the simplest form, the projection can take place with the help of a hole, but in most cases lenses are used.

Cameras can be used as a basis of diverse 3D imaging methods. A single camera can be enhanced with a special light source that provides a consistent illumination. From how the shades form on the object under different angles, its 3D form can be calculated. A single camera can also be used for 3D imaging with the focus technique, where multiple images are taken from the object of interest with various focusing distances [3]. For each pixel, the focusing distance, when it appears to be the sharpest, is considered as its distance to the sensor.

When two cameras are placed next to each other at a known distance, and the corresponding images' differences are used to calculate depth data, we are talking about a stereoscopic camera. Since this approach is feature-based, it requires the object of interest to have a distinguishable texture or feature such as a known dimension [4]. A stereoscopic camera is not capable of detecting homogeneous surfaces on its own. To enhance such a device, a point-matrix projector can be used, which illuminates the scene with a point matrix that provides enough features. This usually takes place in the infrared spectrum, so that the projected pattern remains invisible to the human eye. Taking this approach one step further, a known random point pattern can be used, as in the first version of the Kinect sensor (Kinect for Xbox 360). This way, the detected image can be compared to the known pattern, and distances can be calculated from the displacements. The stereoscopic approach can be extended into a rig where multiple cameras are placed on a frame, facing towards the centre. This setup provides images of an object from multiple known positions and angles, which can be fed into an algorithm that—similarly to stereoscopy—calculates the 3D representation. If even the position and orientation (together called pose) of the cameras are not known, a 3D point cloud can still be calculated with the help of the so-called photogrammetry approach [5,6]. This technique also utilises an algorithm that finds corresponding features in the images and calculates the cameras' pose. Using unconstrained image sets for 3D reconstruction has a use-case in landmark-surveying, where photos posted on social media can be fed into a photogrammetry algorithm [7]. This approach democratises the collection of valuable data and reduces the need for manual data acquisition, which would normally take place with aerial photography or using special scanner systems, for example as presented in [8]. Advanced 3D reconstruction techniques from posed RGB images include approaches where convolutional neural networks (CNN) are used to extract features from the images before backprojecting and accumulating them into 3D points and letting another CNN to refine the 3D features [9]. Besides the difficulties

with reconstructing homogeneous surfaces by certain camera-based techniques, they generally fail to reconstruct non-Lambertian surfaces, i.e., transparent and reflective ones too. To overcome this, Sajjan et al. developed ClearGasp [10], a machine learning algorithm capable of reconstructing transparent objects from RGB-D images.

Thanks to the constant improvement of electronics hardware and of the associated software, many technologies are becoming available to a widening user base. Traditionally, expensive equipment was needed for obtaining 3D information of objects in the context of various scientific fields, including but not limited to archaeology [8], architecture [11], geoinformatics [12], engineering [13] and design. Thanks to the improvement of camera-based 3D techniques, 3D imaging is no longer solely in the hands of a few specialists. The focus of this paper is set on techniques that provide cost-effective 3D imaging solutions. These enable users from new application areas as well as non-professionals to use 3D data for their benefits. Firstly, an active stereoscopy-based mapping technique, the so-called Real-time appearance-based mapping method, and then a technique derived from smartphone augmented reality technology are reviewed. Following this, several use-case scenarios are presented, discussing the usability of each technique. Finally, a qualitative comparison between the two methods is provided.
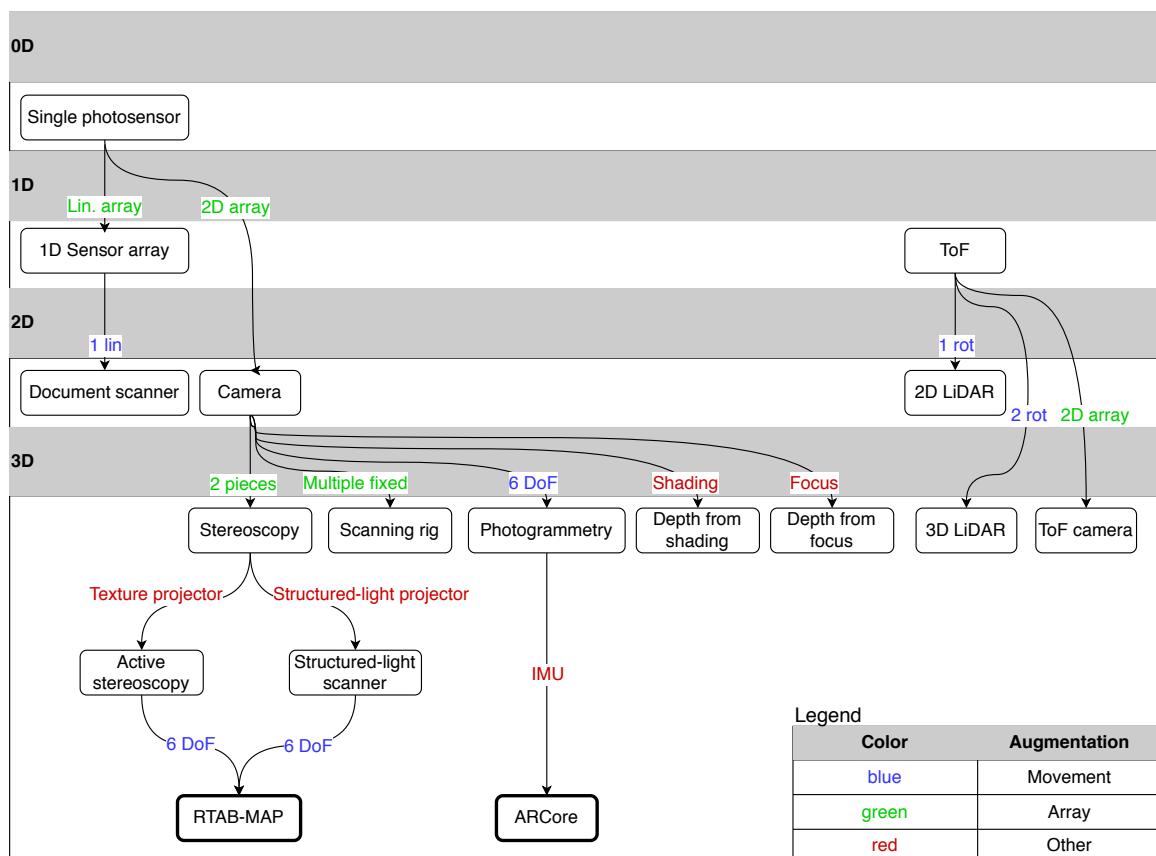

Figure 1. A summary of optical sensors.

2. Overview of the Investigated Technologies

2.1. ARCore

AR means that a scene captured by a camera is enhanced by overlaying dynamic 3D models on the image. Augmented reality applications are used across a wide variety of fields. With its help, visualisations can be created for educational purposes, where students can explore 3D models and animations in an immersive manner [14,15]. Hanafi et al. provide a comparison between various AR software development kits (SDK) in the context of an educational application in the chemical field [16]. Commercial application fields include applications for placing the models of products in the user's

environment. According to experimental studies, this can reduce the consumers' cognitive load during the planning and product selection phase [17]. Entertainment applications include immersive games that provide the user with an experience where characters and other interactive objects are present in the user's environment. However, according to the study of Wölfel et al. [18], technical factors still limit the overall increase in user experience in comparison to non-AR gaming.

Besides specialised AR hardware, such as Microsoft's HoloLens [19] or Google's Glass [20], augmented reality has been available on smartphones since their introduction in the early 2000s [21]. Since then, the smartphone industry has become dominated by Apple and Google, as far as operating systems go. Both companies provide their own AR SDKs with the purpose of giving developers a framework to implement AR applications on their platforms. In this paper, Google's AR SDK, the so-called ARCore [22], is discussed in detail, focusing on 3D reconstruction with smartphones featuring no AR-specific hardware.

From the technical perspective, AR requires three key capabilities: Motion tracking, Environmental understanding, and Light estimation [22]. Motion tracking means that the device's 6 DoF pose is to be detected relative to its environment. In the simplest case, a smartphone AR application utilises the camera stream for feature-based pose estimation, which is enhanced by the orientation and acceleration data provided by the embedded inertial measurement unit (IMU). In advanced, AR-specific smartphones, special depth sensors can be present, such as stereo cameras as in Google's now discontinued Tango project or ToF sensors in certain Android phones. Light estimation enables the lighting of the virtual objects to be adapted to the environment's conditions in order to provide a more realistic experience. Besides the pose of the device, a 3D reconstruction of the environment is also desirable for being able to place augmented objects on various surfaces as well as to let a real-word object occlude a virtual object. In this paper, the utilisation of the 3D reconstruction provided by ARCore is reviewed.

As mentioned above, Google discontinued its Tango project, which was succeeded by ARCore, a universal AR SDK and framework, which does not require any special hardware, such as the depth sensors in Tango-enabled devices. The discontinuation of Tango also meant that the corresponding 3D reconstruction application, Google Constructor, was revoked. Since then, there is no publicly available official 3D reconstruction solution from Google. To overcome this, Vonásek [23] implemented Tango technology with utilising ARCore and brought it to commercial Android phones. Vonásek previously worked on a similar application for Tango, which served as a basis for the ARCore-based app. The application uses ARCore to get the device pose and feature points, from which it deduces depth data. According to the developer, although the Tango3DR library is deprecated, it is still the most advanced solution for meshing, thus it was not replaced yet. As Google's AR technology advances, and developers are given access to more and more features, the 3D Scanner for ARCore is constantly evolving. In this paper, the usability of the version that was available for non-ToF phones at the time of conducting the case studies is discussed.

3D Scanner for ARCore enables the user to obtain a 3D reconstruction of an environment by walking around with the phone, pointing the camera at the objects of the scene. The app constructs a simple mesh in real-time, which is overlaid on the camera stream for instant feedback. The user can select from various presets, including one for indoor, one for outdoor, and one for face reconstruction. The resolutions also vary respectively, ranging from 2 to 8 cm. When the user is finished with scanning, post-processing takes place, starting with the optional Poisson reconstruction, where holes in the mesh are closed to form watertight geometries. Following this, the models are merged, after which the geometry is simplified. Finally, a texture is produced from the photos and the mesh is converted to the OBJ file format. The app also features a simple 3D viewer, which also enables viewing the generated meshes in virtual reality.

2.2. RTAB-Map

The main function of RTAB-Map is RGB-D or LiDAR-based SLAM (Simultaneous Localisation and Mapping), but since it generates a 3D representation of the environment, it can also be used for

3D reconstruction. It is a three-dimensional, graph-based approach that detects occurrences when an image comes from a previously seen location. When a loop closure is detected, a constraint is added to the graph and the error is minimised [24,25]. To capture RGB-D data, an Intel® (Intel Corporation, Santa Clara, CA, USA, 2019) RealSense™ Depth Camera D435 was used. The camera works similarly to a Microsoft Kinect: it has a point matrix projector, two infrared cameras, and an RGB camera. The calculation of the depth information is performed onboard the camera, and through a wrapper [26] it provides ROS with RGB-D data. ROS stands for Robot Operating System, which is a widely used open-source robot software framework. It provides tools and libraries for obtaining, building, writing, and running code across multiple computers. The RTAB-Map package implements odometry and mapping and provides a visualisation tool with which the resulting point clouds can be exported in their raw or processed form into meshes. Running RTAB-Map for SLAM in ROS environment can export the captured point cloud to PCD format.
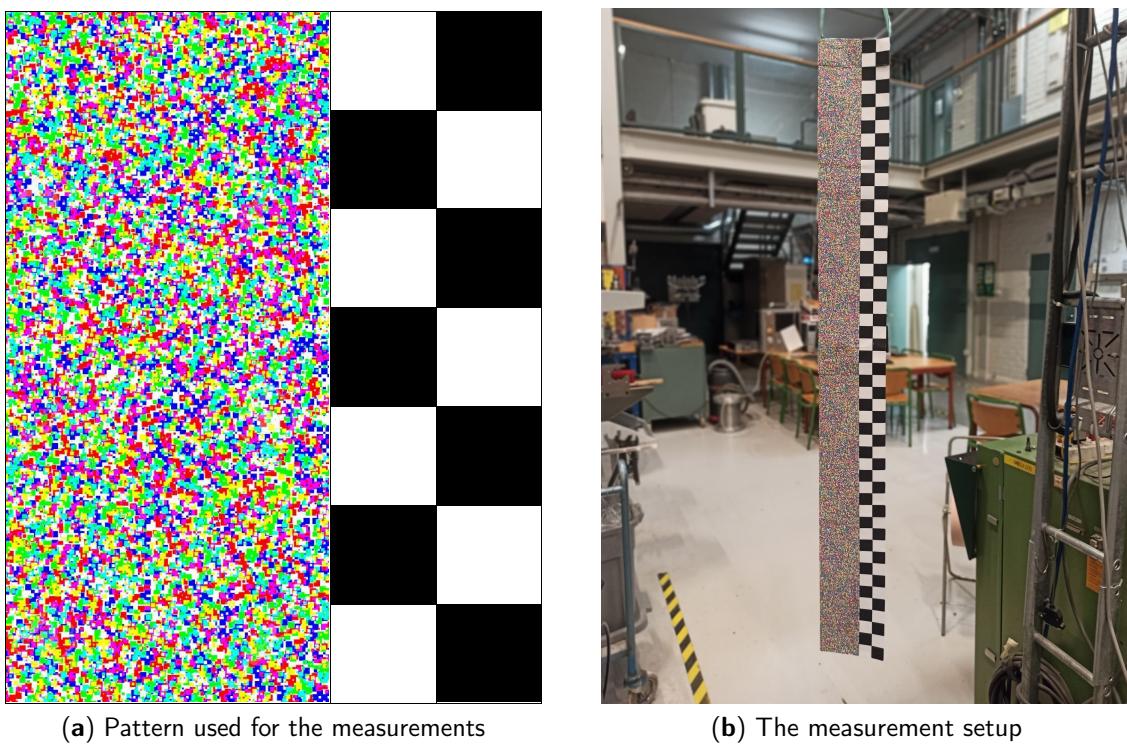
The RealSense™ (Intel Corporation, Santa Clara, CA, USA, 2019) D435, which belongs to the category of active stereoscopy, is equipped with a built in IMU. Combined with RTAB-MAP for SLAM, it is possible to achieve mapping and localisation. The built-in IMU can only provide reliable pose data for a short time due to a runtime-related drift error in the sensors. Therefore, moving the device too fast or too suddenly can interrupt the recording process and result in a faulty point cloud.

3. Experimental Comparison

3.1. Methodology

Surveys and comparisons of various 3D perception technologies usually follow similar methodologies. As such, Fürsattel et al. [2] provide a comparison of recent ToF cameras in regard to systematic errors by establishing a benchmarking framework. The analysis considers factors such as the warm-up time, temporal noise, amplitude-related distance error, wiggling, and the effect of various settings. Giancola et al. [1] surveyed various 3D cameras along similar aspects, including temperature stability, pixel-wise range measurement, the level of uncertainty and systematic error related to pixel position, the effects of incidence angle on the target, as well as the material of the target. The survey includes ToF, structured light, and active stereoscopy, highlighting the strengths and weaknesses of each technique. The subject of these works, however, are all fixed-frame imaging techniques. These deliver depth images in a known coordinate system, in which the ranges can explicitly be determined. In contrast, both in the case of RTAB-Map and ARCore, the resulting point cloud or mesh is generated based on images (RGB or RGB-D) from multiple angles, i.e., from different coordinate frames. This means that the dimensions cannot be explicitly defined, but the measurement object has to be segmented, and its pose has to be defined. To reduce this problem, the presented approach took advantage of the fact that the coordinate frames are still placed approximately where the measurement was started, i.e., in front of the measurement object.

Both the RTAB-Map and the ARCore technologies use the IMU signal of the given device and, by fusing the obtained orientation data with the content of the captured images, they can build a model of the scanned object with an approximately appropriate scale factor. This scale factor specifies the size relationship between the created model and the actual object. To provide a qualitative comparison between the two 3D reconstruction technologies, a simple methodology was applied to measure the one-dimensional length of a reference object. To maximise the detectability of the test object, a random colour noise pattern was used along with a chequerboard scale, as shown in Figure 2a. The test object was suspended in a way that from the perspective of where the scanning took place no object would be visible within the range of the imaging devices. This enabled both of the feature-based algorithms to reconstruct the test object with a minimum amount of points detected from the environment. The test object was scanned with each technique twenty times, then fed the resulting point clouds into a custom-implemented processing script to measure the length of the test object. In the script, which was implemented in MATLAB, the pointCloud object of the Computer Vision Toolbox was utilised.

(**a**) Pattern used for the measurements

(**b**) The measurement setup

Figure 2. The reference object—random colour noise pattern along with a chequerboard scale.

For the discussion of the algorithm, a coordinate system is assumed, the origin of which is at the initial camera pose, the x-axis is horizontal and points to the right, the y-axis is vertical and points upwards, whereas the z-axis is horizontal and points towards the camera from the object, as shown in Figure 3. As the first step, the script removes outliers along the z-axis, which means that most objects that were picked up from the background get ignored. Following this, the projection of the point cloud to the xy plane is used to find the orientation of the test object. A random sample consensus (RANSAC) algorithm finds the most dominant line in the point cloud, which is assumed to correspond to the length of the test object, as Figure 4a shows. Then, the angle of this line is used to rotate and move the point cloud so that the x-axis aligns with the length of the test object. As shown in Figure 4b, a histogram is then created to determine the distribution of the detected points along the x-axis. A threshold is defined by calculating the average of the non-zero bins. The points are iterated through from both ends along the x-axis to find the end of the test object by comparing the bin values to the threshold. The measured length is calculated by subtracting the x value of the lower limit from the x value of the upper limit. Besides keeping the focus on RTAB-Map and ARCore, point clouds created with the single-frame measurement mode of the RealSense camera are also processed and evaluated. The MATLAB script, which was written for processing and evaluating the point clouds and meshes according to the above-described methodology, can be found at the open repository: http://github.com/wlfdm/3d-scanning.
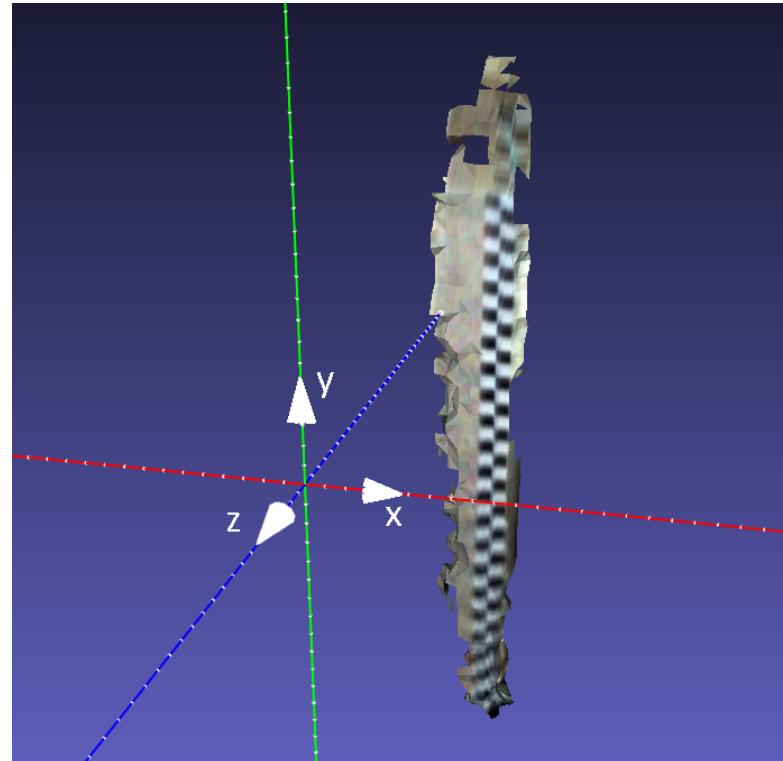
Figure 3. A mesh of the test object scanned with ARCore with the coordinate system.

(**a**) Projection of the point cloudto the XY plane

(**b**) Histogram along the rotated *x*-axis
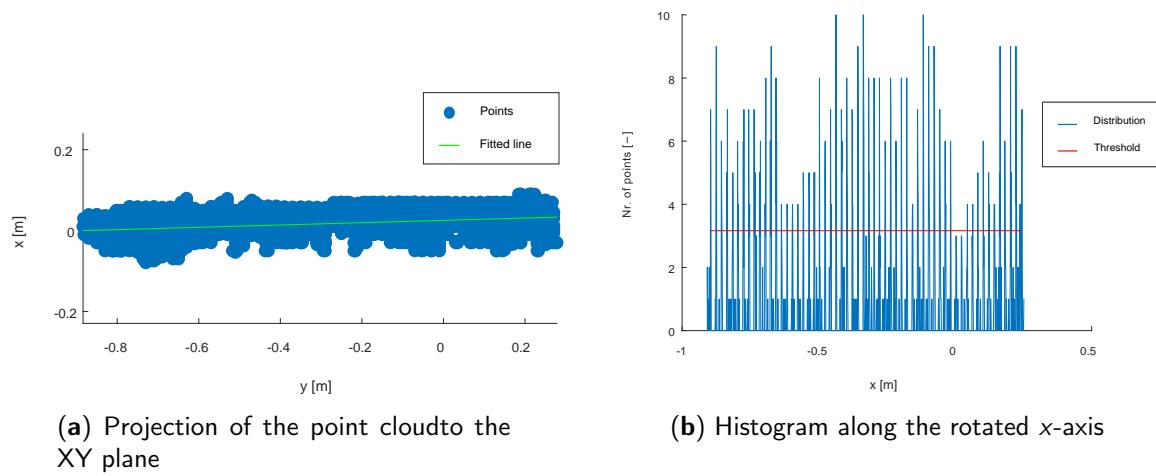
Figure 4. Processing the point clouds.

3.2. Results and Discussion

Table 1 summarises the results of the reference measurements, whereas Figure 5 provides a visual representation of the measured values.

Statistical analysis of the data sets was performed by means of calculating the following values.

The mean or average x_{avg} can be assumed to be the best measured value, based on the set of measurements:

$$x_{avg} = \frac{x_1 + x_2 + ... + x_N}{N} \tag{1}$$

The range or spread R of the data set is the difference between the maximum and the minimum value of the data set:

$$R = x_{max} - x_{min} \tag{2}$$

Standard deviation of the mean σ_{avg} is the range around x_{avg} within the actual value of x will lie:

$$\sigma_{avg} = \frac{\sqrt{\frac{(\sum(x_i - x_{avg})}{n}}}{\sqrt{N}} \tag{3}$$

Measured value x_m is the final reported value of x, which contains both the mean value and the standard deviation of the mean:

$$x_m = x_{avg} \pm \sigma_{avg} \tag{4}$$

It is important to note that there are multiple sources of systematic errors in both scanning technologies. As such, a tendency for underestimation of lengths can be observed due to the loop closure feature potentially shifting the parts of the scan overlapping each other. Apart from that, a shorter measured length can also be caused by incomplete meshes, where the end section of the reference object was not detected. On the other hand, a longer measured length can occur when parts of the environment are being captured, such as the string that was used for suspending the reference object. Generating the point cloud in single depth capture mode eliminates the first two types of errors that could cause a shorter measured length. Accordingly, as can be seen in Figure 5 and in Table 1, the single depth capture delivered solely overshooting results. The generated meshes and point clouds along with the table containing the results can be found at the open repository: http://github.com/wlfdm/3d-scanning.
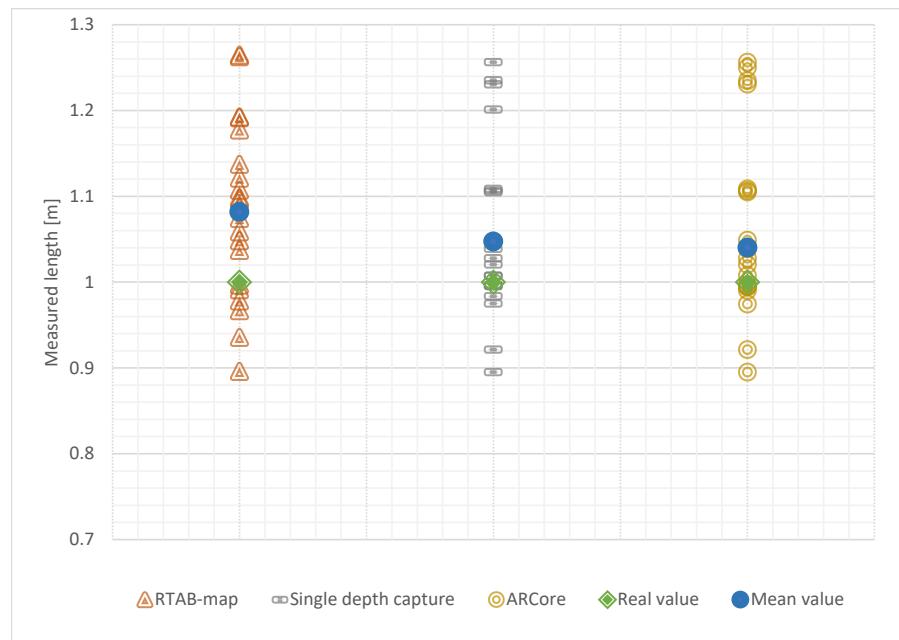
Figure 5. Plot of the measured values.

Table 1. Results of the reference measurements.

| Technology	RS		ARCore
	RTAB-map	Single	
	1.2627	1.0388	1.2498
	1.0368	1.2303	1.2564
	1.1372	1.2011	1.0279
	1.1211	1.0074	1.2347
	1.1071	1.1047	0.9744
	1.2646	1.0073	1.2306
	1.0741	0.9951	0.9946
	0.8959	0.9956	0.9902
	1.0978	0.9835	1.105
Measurements	0.9666	0.9971	1.0077
	1.1771	0.895	0.9958
	1.1915	1.0487	0.9971
	0.9356	1.0205	0.9945
	0.991	1.1086	0.895
	1.0481	1.1065	1.0205
	1.0585	0.9213	1.0388
	0.9965	1.2564	1.0487
	1.1926	1.0279	1.1086
	0.9781	1.2353	1.1065
	1.0952	0.9749	0.9213
Mean	1.081	1.058	1.060
Range	0.369	0.361	0.361
Std of the mean	0.023	0.023	0.024
Measured value	1.081	1.058	1.060
+/−	0.023	0.023	0.024

## 4. Case Studies

4.1. Surveying in Laboratory Automation

The above-mentioned technologies were tested in the context of an ongoing research project, the subject of which revolves around studying the usability of various new technologies in laboratory automation [27,28]. Laboratory automation as a field of research addresses technologies, the aim of which is to automate the processes in various research and development laboratories in life sciences, ranging from the academia through healthcare to pharmaceutical companies. Such technologies include separated devices that are capable of performing a certain task autonomously, such as liquid handler robots, storage units, readers, and various analytic devices. However, in laboratory automation, the ultimate goal is to integrate the partly automated subprocesses into a comprehensive overlaying workflow by providing interfaces and a control system. Approaches that are considered ubiquitous in other industries, such as the application of robots for transport purposes, are just beginning to be widespread in life science laboratories [29]. Similarly, the application of new technologies that were previously only applied in special contexts, such as virtual and augmented reality or 3D reconstruction, are also beginning to find their way in automated laboratories. As such, in this chapter, a use-case for 3D reconstruction is presented, where the laboratory presented in Figure had to be surveyed for planning, visualisation, and simulation purposes. For this, both the ARCore-based and the RTAB-Map-based approaches were tested. The usability of the resulting meshes highly depends on the applied technology, since factors such as size- and shape accuracy as well as the consistency of the meshes play a big role.

Firstly, the ARCore-based 3D Scanning application was tested. It is important to mention that, for this, the version 04/2019 of the application was used, and since then several improvements were implemented by the developer—among others—on the meshing and on the position accuracy. Figures 6b and 7a present the resulting textured mesh, which has many missing areas, especially at

homogeneous or reflecting surfaces. On the contrary, feature-rich surfaces, such as the tabloids on the wall, are well preserved, and the absolute accuracy of the dimensions of the resulting models is also relatively high. Measuring distances of specific points on the mesh and comparing it to values from ground plans and actual measurements showed that the relative error lies under 1%. These properties altogether make the scanned mesh insufficient for direct use in simulation but sufficient to provide a guide for the manual modelling.

(**a**) Picture of the laboratory　　　　　　　　　　　　(**b**) Detail of the ARCore mesh
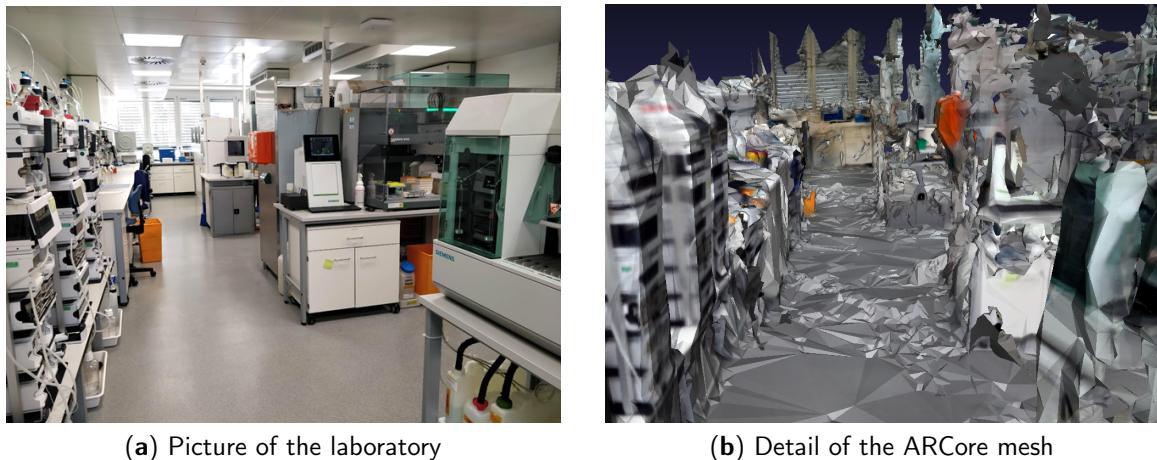
Figure 6. The laboratory in real life and as scanned.

(**a**) Mesh created with ARCore　　　　　　　　　　　　(**b**) Mesh created with RTAB-MAP
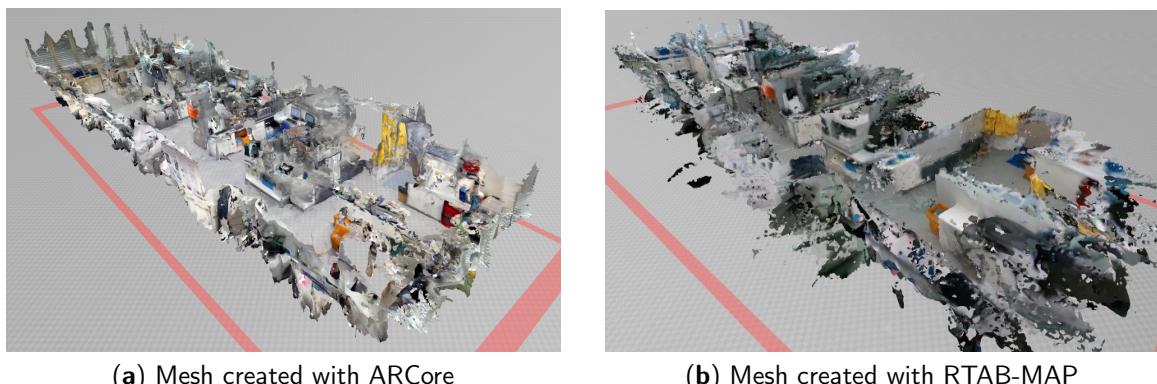
Figure 7. Mesh of the laboratory.

Another 3D scanning technology was tested with an Intel RealSense depth camera [30] and the Real-Time Appearance-Based Mapping (RTAB-Map) [31], an RGB-D SLAM implementation for ROS. As can be seen in Figure 7, the mesh proved to be more continuous than the one created with ARCore, despite the fact that the scanning time was significantly shorter. This can be due to the fact that the RealSense camera also has a point matrix projector, which provides enough texture for otherwise homogeneous surfaces. On the other hand, it can also be observed that more falsely detected points appear, i.e., points that are not part of any objects but "float" in the air. The reason for such errors can be reflections or other optical artefacts.

For the RTAB-Map technology, another potential use-case was identified in the context of the above-mentioned laboratory automation project. The main focus of this project is namely to research the usability of a mobile robot for sample transportation and other tasks in the laboratory and to develop novel technologies and applications in this context. As such, a mobile robot needs a means of localising and navigating itself in its environment. In the case of ground-bound robots, this localisation has to take place in three DoF (two translations along the floor and a rotation around the vertical axis). For this purpose, usually the so-called simultaneous localisation and mapping technique is used.

In most of the cases, the algorithm uses a 2D point cloud delivered from a laser scanner and creates a map of the premises, where the robot operates. These data are enhanced with the angular position data of the wheels delivered from the wheel encoders and optionally with orientation and acceleration data delivered from an on-board IMU. However, if a robot is not ground-bound, such as a drone, localisation in six DoF is required (three translations and three rotations). As an outlook, this scenario was considered and the localisation capabilities of the RTAB-Map algorithm were tested.

For this purpose, the same Intel RealSense camera was used for the scanning. Figure 8 shows the output of the RTAB-Map Visualizer while navigating in a previously created map of the laboratory. On the bottom left, the feature points detected on the camera image are marked, while on the right the path of the mapping session and the current pose of the camera can be seen. On the top left, the projected two-dimensional map and the position of the camera are presented in the ROS Visualizer (RViz). It is important to note that this map must be processed by hand by removing the falsely detected obstacle points before using the map with an actual robot or an autonomous vehicle.
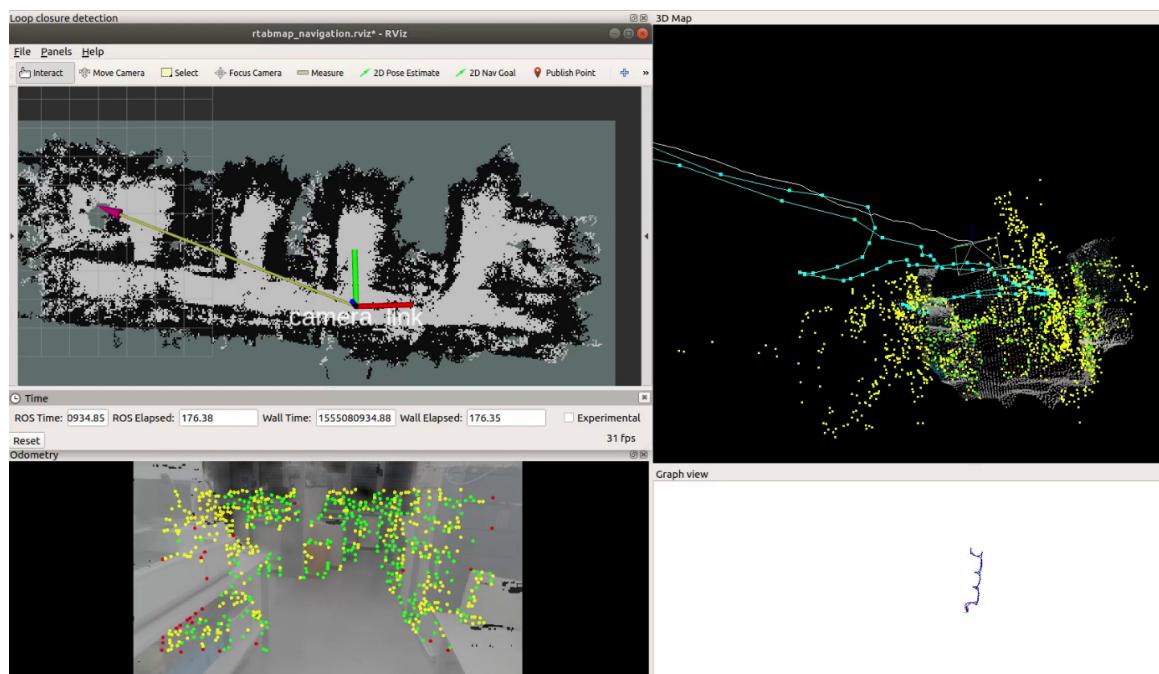
Figure 8. Output of the RTAB-Map Visualizer during navigation. Colour codes: Cyan and Blue = trajectory, Green = Inliers, Yellow = Not matched features from previous frame(s), Red = Outliers.

4.2. Landmark Surveying

In another case study, the discussed techniques were tested for landmark surveying. In this application, a cave was measured using both technologies. This application was suitable for the scale display of the cave passages, and, based on measurements made in the vicinity of the cave entrance, the model could be placed in the Google Earth environmental model, which served as a reference for placing the cave interiors in relation to the external environment. As Figure 9 shows, the mesh created with the ARCore Scanner has more discontinuities as the one created with RTAB-Map. However, with ARCore, the whole length of the cave could be scanned without interruption, whereas RTAB-Map lost the reference after approximately ten meters. It is important to note here that the Intel RealSense version without an IMU was used, whereas the inertial information apparently gives an advantage to ARCore in regard to motion tracking. The RTAB-Map mesh being more continuous is due to the fact that the RealSense camera features a point matrix projector, which provides an artificial texture for homogeneous surfaces, such as the light grey clay walls of the cave.

(**a**) Mesh created with ARCore　　　　　　　　　　(**b**) Mesh created with RTAB-MAP
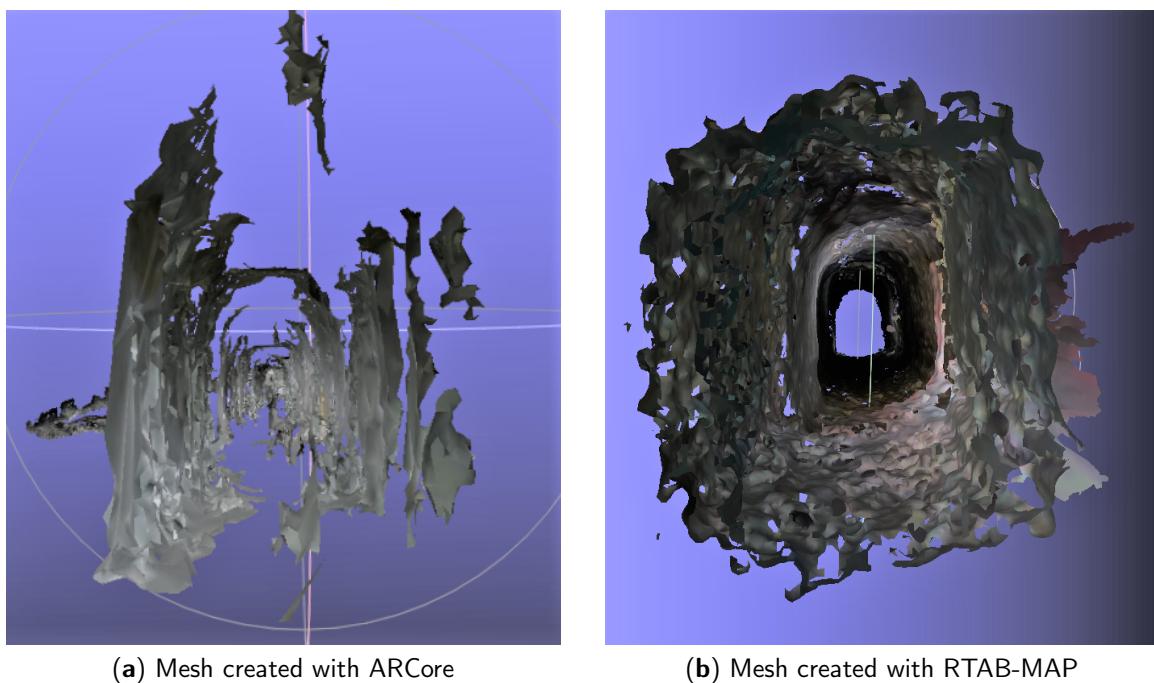
Figure 9. Meshes of the cave.

5. Conclusions

This paper aimed at investigating cost-effective, easy-to-access 3D reconstruction technologies. One can see that these technologies alone cannot replace more advanced 3D scanning apparatus, i.e., Time of Flight (ToF) LiDARs, but they can be utilised in various use-cases where lower quality but quickly-generated models are applicable.

Based on the experimental work presented in this study, we can conclude that the shape and direction accuracy of the resulting point clouds are acceptable in most practical situations. However, the meshes are often non-continuous, partly due to a systematic error caused by the reflection anomalies of the captured surfaces. Due to these factors, the so obtained results are not suitable for high-quality visualisation purposes, but they can provide a good basis for planning, surveying, and further manual processing. The presented case studies justify that, using the RTAB-Map, ARCore, or other similar techniques, the preparation of 3D scans takes a relatively short time. This makes the approach well suited for the applications where the simplicity of the devices and the fast incremental model generation are preferred over the geometric accuracy and model quality. Concerning the quality-related issues regarding both investigated techniques, it can be concluded that the result of the measurement procedure is not deterministic and the resulting models suffer from severe mesh continuity problems. The present study was conducted in the context of a laboratory automation and robotisation project, which was presented as a representative use-case. In this regard, the discussed technologies provide a useful way of facility surveying and draft environment model capture for system design purposes.

For further evaluating the performance of each technology, reference measurements may be conducted with a sensor of higher accuracy, such as a LiDAR.

Abbreviations

The following abbreviations are used in this manuscript:

Abbreviation	Name
0D	Null-dimensional
1D	One-dimensional
2D	Two-dimensional
3D	Three-dimensional
AR	Augmented reality
CAD	Computer aided design
DoF	Degree of freedom
IMU	Inertial measurement unit
LiDAR	Light detection and ranging
OBJ	Object file extensions
RANSAC	Random sample consensus
RGB	Red Green Blue
RGB-D	Red Green Blue Depth
ROS	Robot operating system
RTAB-Map	Real-time appearance-based mapping
Rviz	ROS Visualizer
SDK	Software development kit
ToF	Time-of-flight

References

1. Giancola, S.; Valenti, M.; Sala, R. *A Survey on 3D Cameras: Metrological Comparison of Time-of-Flight, Structured-Light and Active Stereoscopy Technologies*; Technical Report; SpringerBriefs in Computer Science; Springer: Berlin/Heidelberg, Germany, 2018.
2. Fursattel, P.; Placht, S.; Balda, M.; Schaller, C.; Hofmann, H.; Maier, A.; Riess, C. A Comparative Error Analysis of Current Time-of-Flight Sensors. *IEEE Trans. Comput. Imaging* **2015**, *2*, 27–41. [CrossRef]
3. Kulkarni, J.B.; Sheelarani, C.M. Generation of depth map based on depth from focus: A survey. In Proceedings of the 2015 1st International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 26–27 February 2015; pp. 716–720. [CrossRef]
4. Ekberg, P.; Daemi, B.; Mattsson, L. 3D precision measurements of meter sized surfaces using low cost illumination and camera techniques. *Meas. Sci. Technol.* **2017**, *28*, 045403. [CrossRef]
5. Schönberger, J.L.; Zheng, E.; Pollefeys, M.; Frahm, J.M. Pixelwise View Selection for Unstructured Multi-View Stereo. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016.
6. Schonberger, J.L.; Frahm, J.M. Structure-from-Motion Revisited. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 4104–4113. [CrossRef]

7.   Martin-Brualla, R.; Radwan, N.; Sajjadi, M.S.M.; Barron, J.T.; Dosovitskiy, A.; Duckworth, D. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. *arXiv* **2020**, arXiv:2008.02268.

8.   Molnár, A. *Surveying Archaeological Sites and Architectural Monuments with Aerial Drone Photos*; Technical Report 7; Department of Applied Informatics, John von Neumann Faculty of Informatics, Óbuda University: Budapest, Hungary, 2019.

9.   Murez, Z.; van As, T.; Bartolozzi, J.; Sinha, A.; Badrinarayanan, V.; Rabinovich, A. Atlas: End-to-End 3D Scene Reconstruction from Posed Images. *arXiv* **2020**, arXiv:2003.10432.

10.   Sajjan, S.S.; Moore, M.; Pan, M.; Nagaraja, G.; Lee, J.; Zeng, A.; Song, S. ClearGrasp: 3D Shape Estimation of Transparent Objects for Manipulation. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020.

11.   Udvardy, P.; Jancsó, T.; Beszédes, B. 3D modelling by UAV survey in a church. In Proceedings of the NTinAD 2019—New Trends in Aviation Development 2019—14th International Scientific Conference, Chlumec nad Cidlinou, Czech Republic, 26–27 September 2019; pp. 189–192; doi:10.1109/NTAD.2019.8875580. [CrossRef]

12.   Jancsó, T.; Engler, P.; Udvardy, P. Aerial Survey Test Project with DJI Phantom 3 Quadcopter Drone; Technical Report. *RevCAD J. Geodesy Cadastre* **2016**, *21*, 59–66.

13.   Kocur, D.; Švecová, M.; Kažimír, P. Determining the Position of the Moving Persons in 3D Space by UWB Sensors using Taylor Series Based Localization Method. *Acta Polytech. Hung.* **2019**, *16*, 2019–2064. [CrossRef]

14.   Molnár, G.; Szűts, Z.; Biró, K. Use of Augmented Reality in Learning. *Acta Polytech. Hung.* **2018**, *15*, 2018–209.

15.   Engelhardt-Nowitzki, C.; Aburaia, M.; Otrebski, R.; Rauer, J.; Orsolits, H. Research-based teaching in Digital Manufacturing and Robotics—The Digital Factory at the UAS Technikum Wien as a Case Example. *Procedia Manuf.* **2020**, *45*, 164–170. [CrossRef]

16.   Hanafi, A.; Elaachak, L.; Bouhorma, M. A comparative study of augmented reality SDKs to develop an educational application in chemical field. In Proceedings of the 2nd International Conference on Networking, Information Systems & Security, Rabat, Morocco, 27–28 March 2019; Part F148154. [CrossRef]

17.   Fan, X.; Chai, Z.; Deng, N.; Dong, X. Adoption of augmented reality in online retailing and consumers' product attitude: A cognitive perspective. *J. Retail. Consum. Serv.* **2020**, *53*, 101986. [CrossRef]

18.   Wolfel, M.; Braun, M.; Beuck, S. How does augmented reality improve the play experience in current augmented reality enhanced smartphone games? In Proceedings of the 2019 International Conference on Cyberworlds (CW 2019), Kyoto, Japan, 2–4 October 2019; pp. 407–410. [CrossRef]

19.   Microsoft HoloLens | Mixed Reality Technology for Business. Available online: https://www.microsoft.com/en-us/hololens (accessed on 7 September 2020).

20.   Google–Glass. Available online: https://www.google.com/glass/start/ (accessed on 7 September 2020).

21.   Henrysson, A.; Ollila, M. Augmented reality on smartphones. In Proceedings of the ART 2003—IEEE International Augmented Reality Toolkit Workshop, Tokyo, Japan, 7 October 2003; pp. 27–28; doi:10.1109/ART.2003.1320421. [CrossRef]

22.   ARCore overview | Google Developers. Available online: https://developers.google.com/ar/discover (accessed on 7 September 2020).

23.   3D Scanner for ARCore·lvonasek/tango Wiki·GitHub. Available online: https://github.com/lvonasek/tango/wiki/3D-Scanner-for-ARcore (accessed on 9 August 2020).

24.   Labb, M. RTAB-Map as an Open-Source Lidar and Visual SLAM Library for Large-Scale and Long-Term Online Operation. *J. Field Robot.* **2019**, *36*, 416–446. [CrossRef]

25.   RTAB-Map. 2019. Available online: http://introlab.github.io/rtabmap/ (accessed on 6 September 2020).

26.   Intel(R) RealSense(TM) ROS Wrapper for D400 Series, SR300 Camera and T265 Tracking Module: IntelRealSense/Realsense-Ros. 2019. Available online: https://github.com/IntelRealSense/realsense-ros (accessed on 6 August 2020).

27.   Wolf, A.; Széll, K. A review on robotics in life science automation. In Proceedings of the AIS 2019 14th International Symposium on Applied Informatics and Related Areas Organized in the Frame of Hungarian Science Festival 2019 by Óbuda University, Székesfehérvár, Hungary, 14 November 2019; pp. 106–111.

28.   Wolf, A.; Galambos, P.; Széll, K. Device integration concepts in laboratory automation. In Proceedings of the 2020 IEEE 24th International Conference on Intelligent Engineering Systems (INES), Reykjavík, Iceland, 8–10 July 2020.

29.   Fleischer, H.; Thurow, K. *Automation Solutions for Analytical Measurements: Concepts and Applications*; Wiley-VCH: Weinheim, Germany, 2017; p. 272.

30. Overview of the Intel® RealSense™ Depth Camera. 2019. Available online: https://software.intel.com/en-us/realsense/d400 (accessed on 9 August 2020).
31. rtabmap_ros—ROS Wiki. 2019. Available online: http://wiki.ros.org/rtabmap_ros (accessed on 9 August 2020).

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.