

# IMPLEMENTATION AND FIRST EVALUATION OF AN INDOOR MAPPING APPLICATION USING SMARTPHONES AND AR FRAMEWORKS

O. Hasler, S. Blaser, S. Nebiker

Institute of Geomatics Engineering, FHNW University of Applied Sciences and Arts Northwestern Switzerland, Muttenz, Switzerland – (oliver.hasler, stefan.blaser, stephan.nebiker)@fhnw.ch

## Commission II

**KEY WORDS:** Augmented Reality, Indoor, Smartphone, Mobile Mapping

### ABSTRACT:

In this paper, we present the implementation of a smartphone-based indoor mobile mapping application based on an augmented reality (AR) framework and a subsequent performance evaluation in demanding indoor environments. The implementation runs on Android and iOS devices and demonstrates the great potential of smartphone-based 3D mobile mapping. The application includes several functionalities such as device tracking, coordinate, and distance measuring as well as capturing georeferenced imagery. We evaluate our prototype system by comparing measured points from the tracked device with ground control points in an indoor environment with two different campaigns. The first campaign consists of an open, one-way trajectory whereas the second campaign incorporates a loop closure. In the second campaign, the underlying AR framework successfully recognized the start location and correctly repositioned the device. Our results show that the absolute 3D accuracy of device tracking with a standard smartphone is around 1% of the travelled distance and that the local 3D accuracy reaches sub-decimetre level.

## 1. INTRODUCTION

The demand for capturing accurate 3D information is growing in a wide variety of disciplines such as BIM (Building Information Modelling), facility management or indoor navigation. Until recently, mapping indoor environments was a demanding task, requiring highly specialized multi-sensor systems such as terrestrial or mobile laser scanners. Then, new high quality indoor mobile mapping systems (MMS) were introduced such as the BIMAGE backpack (Blaser, Nebiker, & Cavegn, 2017). With such a backpack, 3D point clouds and highly detailed 3D image spaces (Nebiker, Cavegn, & Loesch, 2015) of large buildings, construction sites or tunnels can be captured. However, when it comes to keeping the data up-to-date, using such high-end MMS would be too costly and the system would be restricted to a small group of experts. Hence, there should be a simple and cost-effective solution allowing building owners or facility managers to keep the digital twin of their infrastructure up-to-date.

In recent years, the computing capacity of mobile devices has rapidly increased, which is enabling more and more computing intensive applications. A typical example are Augmented Reality (AR) applications, which are very demanding with respect to scene tracking and augmentation in real-time, tasks which were not possible on mobile devices a decade ago. Since Niantic has released Pokémon Go in 2016, the number of AR applications is rapidly increasing. Although Pokémon Go was a geospatial AR application, the most common AR applications are placing virtual 3D objects in an arbitrary scene using either a smartphone or AR glasses. These 3D objects can be as simple as a toy figure or as complex as a scaled 3D city model. Most often, these AR applications are restricted to work only in a single room or a small area.

With the introduction of the AR frameworks ARCore (Google, n.d.) and ARKit (Apple Developers, 2019), developing AR applications has been greatly simplified. These AR frameworks support device motion tracking and scene understanding. Visually distinct features from the camera image – called

feature points – combined with inertial measurements from the device's IMU are used to calculate the device's pose relative to the environment. Clusters of feature points that lie on horizontal or vertical surfaces such as tables or walls are detected as planar surfaces. Both ARCore and ARKit require mobile devices with calibrated cameras and the generated point cloud is at world scale.

At the Institute of Geomatics at the FHNW, we are developing a new AR mapping application, which shall combine the advantages of the local tracking of an AR framework with referencing the device to a reference image dataset, which has been georeferenced in a geodetic reference system (Nebiker et al., 2015). As a first step, we developed an application, which is able to motion track the device, measure points, localize itself to a specific reference system and capture photographs with absolute orientation. Once it is possible to align the captured photographs to a georeferenced image database, this application is ready for various mapping tasks with high global accuracy.

Our paper is structured as follows: in chapter 2, we discuss the related work. In chapter 3, we describe our development and architecture and in chapter 4 and 5, we outline our accuracy experiments and their result. Finally, in chapter 6 we give a conclusion and an outlook to future developments.

## 2. RELATED WORK

There are different types of indoor mapping systems. On the one hand, there are static mapping systems like terrestrial laser scanners, which scan the environment with high precision at the cost of a time-consuming data collection process. On the other hand, indoor mobile mapping systems (MMS) are becoming more popular since the data collection can be performed while driving or walking through the environment. Different indoor MMS have been proposed, which can be categorized either by the platform type or by the sensors used. Generally, there are backpack-based systems, handheld systems and trolley-based systems. The Würzburg backpack (Lauterbach et al., 2015), for example, consists of a 2D laser profiler and a Riegl VZ-400 laser scanner. Another backpack system (Blaser, Cavegn, &

Nebiker, 2018) combines two Velodyne VLP-16 scanners with a multi-head panorama camera. Among the commercially available backpacks is the Leica Pegasus:Backpack. It uses two Velodyne VLP-16 scanners for the SLAM algorithm and has five divergent rear-facing cameras which are primarily used for visualisation and digitising (Leica Geosystems AG, 2017). Such high-end backpack systems generate accurate 3D information, but they are typically expensive and relatively heavy to carry. Handheld LiDAR-based indoor MMS like the Zebedee (Bosse, Zlot, & Flick, 2012), are easier to carry even in longer mapping campaigns. The Zebedee combines a lightweight laser scanner and an IMU to generate 3D point cloud. Numerous comparable commercial products such as Zeb Revo from GeoSLAM or Stencil from Kaarta (Zhang & Singh, 2017) are available. Kaarta's Stencil uses a Velodyne VLP-16 scanner together with an IMU for point cloud generation. Wheel-based MMS like the NavVis 3D Mapping Trolley has multiple cameras and four laser scanners (NavVis, n.d.). Trolleys like this can be equipped with many sensors but are restricted to flat and obstacle-free ground surfaces. These type of MMS can generate dense and accurate 3D information but geo-referencing is either done in post processing or by measuring ground control points (Lehtola et al., 2017).

For measuring and mapping with consumer devices, there are several AR mapping tools already available. Lenherr (2018) evaluated some of them in terms of functionality and accuracy. Smartphone applications such as CamToPlan or Magicplan (Magicplan, n.d.) allow users to map and create floorplans. However, most available AR applications do not support absolute geo-referencing.

Precise automatic absolute geo-referencing, regardless of the environment, is of great interest for numerous applications. Visual Localization is a promising automatic geo-referencing approach with intense research activity. Sattler et al. (2018) distinguish the following visual localization categories: 3D structure-based, 2D image-based, sequence-based and learning-based localization. In our previous work, we introduced a method for visual localization and pose estimation of a single image to georeferenced RGB-D images (Nebiker et al., 2015; Rettenmund, Fehr, Cavegn, & Nebiker, 2018). This method works in indoor and outdoor environments and does not require ground control points. However, the success of this method depends on ideal conditions such as up-to-date reference images, same lighting, seasonality and similar viewpoint. Since then, new and more robust methods have evolved. DenseSfM proposes a Structure from Motion (SfM) pipeline that uses dense CNN features with keypoint relocation (Widya, Torii, & Okutomi, 2018). Sarlin et al. (2019) also use learned descriptors to improve localization robustness across large variations of appearance. These approaches are more robust than using classical local features like SIFT and its variants and have the potential to solve the absolute image orientation problem. However, these approaches are computationally heavy and at the time of writing, only a few approaches are computing in real-time. Therefore, we have not yet implemented one of these approaches in our AR mapping framework.

### 3. OUR ARCHITECTURE

The main goals of our development included: a simple operation on a broad range of devices, a compatibility with the two most prominent mobile operating systems Android or iOS, and a real-time capability. The minimal functionality should include the possibility to interactively localize the device in an absolute reference frame using control points, to perform point measurements and to capture georeferenced images.

### 3.1 Underlying Frameworks

Our development is based on the widely used game engine Unity. Unity provides a large number of packages, which can be included into a project and Unity-based applications can be deployed to various operating systems. Our project is developed with Unity's AR Foundation package, which includes built-in multi-platform support for AR applications (Unity, 2018). This makes it possible to develop an application, which can run either Google's ARCore or Apple's ARKit depending on the user's device and operating system.

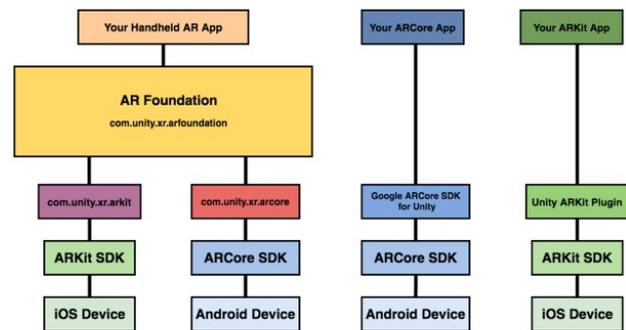


Figure 1. Concept of AR Foundation (Miller, Mowrer, & Weiers, 2018)

### 3.2 Design

Since our development is mainly a proof of concept and not a distribution-ready application, we did not focus on UI and UX aspects in our mobile app development. Instead, the focus was placed on implementing our required functionality, which considerably differs from existing applications. Therefore, all graphical interfaces were implemented with default settings, which allow custom individualisations at a later stage, if required within a distributed application.

### 3.3 Implemented Functionality

**3.3.1 Device Tracking:** The foundation of our application is the device tracking. The underlying AR frameworks support motion tracking by fusing multiple sensors such as accelerometer, gyroscope, magnetometer and camera. Visually distinct feature points from the camera image combined with inertial measurements are used to estimate the device's relative pose to the environment. Furthermore, the framework estimates horizontal and vertical planes with detected feature points, which are mostly located on walls and on the floor.

Once the AR app is started, device tracking starts immediately. The origin of the local AR reference frame coincides with the location where the app was initialised – with the heading of the device defining the direction of the forward axis, the up-axis pointing vertical and the right axis perpendicular to the right. Since either Google's ARCore or Apple's ARKit only run on calibrated devices and multiple sensors are fused, the AR reference frame is automatically at world scale.

**3.3.2 Measurement Functionality:** After the AR app has been initialised, local measurements can be taken immediately. The app supports two measurement modes: point and 3D distance measurements (Figure 2). Other modes such as area, volumetric or slope measurements could additionally be implemented. Both point and distance measurements can either be made directly using individual feature points from the device tracking or using the detected planar surfaces. Measuring on detected planar surfaces has the advantage that measurements can be carried out continuously, even when a surface is lacking visual features.

To execute a measurement, a single tap on the screen on the desired location is required. Depending on the measuring mode, a pop-up window with the local or global coordinates or the 3D distance appears. A 2D distance measurement mode (top down, floorplan) could be implemented additionally, if needed. The coordinates of measured points can be saved to a text file with local and if available global coordinates for further processing.

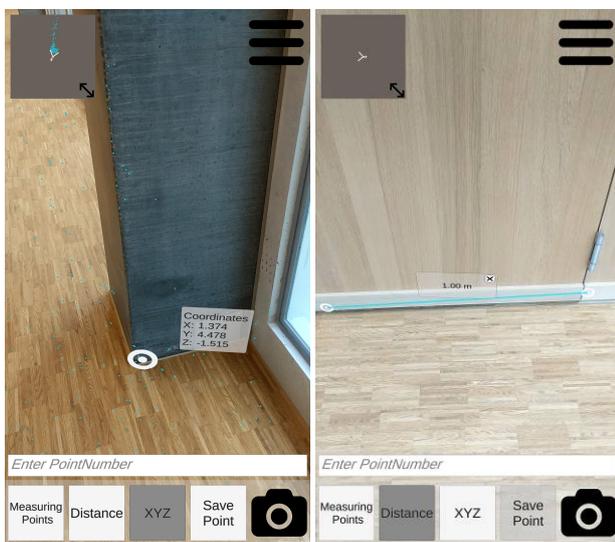


Figure 2. Point (left) and distance (right) measurement modes

**3.3.3 Global Referencing:** For absolute geo-referencing of captured images and for conducting measurements in a global reference frame, the device needs to be related to a reference frame. For a first version, we realized a 6 degree of freedom (6DoF) transformation using ground control points (GCP) in order to transfer the local scene into a global reference frame. To start the referencing process, a list of GCPs can be imported from file. After a successful import, at least three points need to be measured with the AR app and referenced to a GCP by choosing from a dropdown list. Again, measurements can be directly conducted on feature points or on detected planar surfaces. Then, the 6DoF transformation is calculated according to Umeyama (1991) and the residuals are displayed (Figure 3, left). The transformation can easily be evaluated based on residuals and dynamically adjusted by additional point measurements or by the exclusion of points. Once the transformation calculation is correct, any object in the global reference system can be augmented in the camera feed of the app. For verification purposes, the app overlays the GCPs into the camera feed (Figure 3, right).

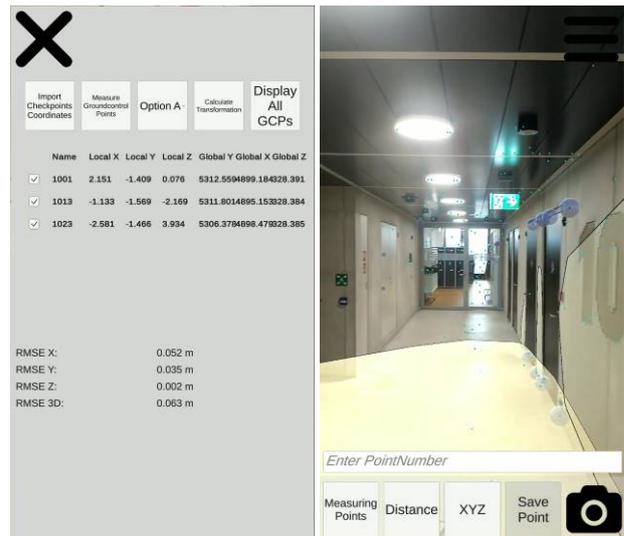


Figure 3. Display of the residuals (left) and verification of the transformation by displaying GCPs as blue spheres (right)

**3.3.4 Photo Capture with Pose:** Finally, it is possible to capture geo-referenced images. Every time a user takes a photo, the app stores the local pose and if available the global pose (position and orientation). With the app, it is also possible to upload the photo with its pose to a web service. For verification purposes, the captured photograph can be displayed in the AR scene at its real location and with its original pose (Figure 4).



Figure 4. Photo display with original pose (captured photo in dashed rectangle)

#### 4. PERFORMANCE EVALUATION

We carried out accuracy evaluations based on 3D point measurements in order to determine the performance and stability of motion tracking and subsequent measuring and mapping accuracy in indoor environments.

In the first experiment, we compare the two 3D measurement methods: feature-point-based and surface-based measurements.

In the following two experiments, we investigate the deviations along multiple trajectories. In the first case, the trajectory describes a route with different start and destination points and in the second case, the trajectory forms a loop. In all of the

following experiments, we used common high-end smartphones such as Google Pixel 2 and Samsung Galaxy S9.

#### 4.1 Test Site

Our test site is located in the new main building of the FHNW Campus in Muttenz/Basel. It covers the eastern part of the 10<sup>th</sup> floor, where four large perpendicular corridors form a loop (Figure 5). The indoor environment of this modern building is challenging since there are a lot of repetitive structures and uniform surfaces including large glass facades. Therefore, extracting distinct visual features is demanding.

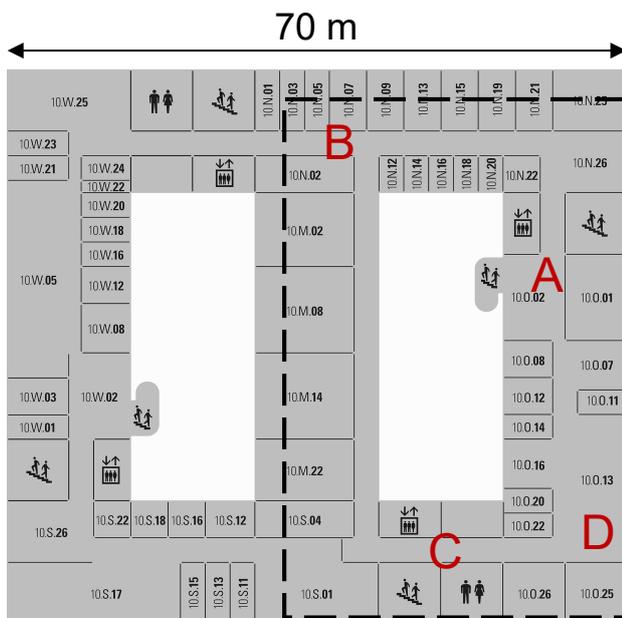


Figure 5. Floorplan of the 10th floor with test area (inside dashed rectangle)

As a reference, we established 137 ground control points (GCPs) which we measured with a multistation Leica MS60. The overall accuracy of the reference system is < 3mm. The GCPs are exactly defined natural points like door corners and intersections between a wall corner and the floor (Figure 6).



Figure 6. Ground control points near location A

#### 4.2 Coordinate Measurement

In the first part of the investigation, we compared different 3D coordinate measurement methods within the AR Scene. The AR framework supports two types of coordinate measurements: 1)

by getting coordinates directly from the raw feature point cloud or 2) by determining the coordinates by intersecting an image ray with a detected surface. By investigating first the quality of the raw feature point cloud and then comparing the residuals between GCPs and measurements on the point cloud and the detected surfaces, we want to know how to measure 3D coordinates for the following investigation.

#### 4.3 One-way trajectory

In the first of the two mapping experiments, we evaluated the tracking performance using a one-way trajectory. This trajectory was created by walking a path from A via B to C (Figure 5). This path is around 90 meters long and includes three direction changes of 90 degrees. Between B and C, there is a long corridor with repetitive structures and large windows as shown in Figure 7. At the starting location A, seven GCPs on different height levels were measured. Along the trajectory to the final location C, nine additional points were measured as checkpoints. The transformation parameters to the global reference frame were derived using the seven GCP measurements at location A. Once the transformation parameters were estimated, all measured points were transformed into the global reference frame and the residuals to the GCPs were calculated.



Figure 7. Repetitive structures in the corridor between B and C

#### 4.4 Closed loop trajectory

In the second mapping experiment, we measured a trajectory, which forms a closed loop from location A via B, C and D and ending again at location A. In this campaign we measured 20 points in total. At location A we measured the identical GCPs and we additionally measured 13 checkpoints along the loop (see Figure 9, right). The trajectory length is about 140 meters and includes four 90-degree turns. We again used the GCPs to calculate the transformation from the local to the global reference frame and then transformed all points with these parameters. Finally, the residuals were calculated for all measured points.

### 5. RESULTS

In this chapter, we show and discuss the results of our three performance investigations. First, we compared two different 3D measurement approaches supported by our application. Second, we examined the tracking quality by performing 3D measurements along two different trajectories, one open and the other closed.

## 5.1 Coordinate measurements

The quality of the raw feature point cloud is rather low because of the high noise. The detected points particularly vary a lot in the direction of the device’s camera. This is not surprising since the 3D points are calculated for every frame and the camera displacement between frames is often only marginal, both in location and orientation. Figure 8 shows the horizontal projection of the resulting point cloud.

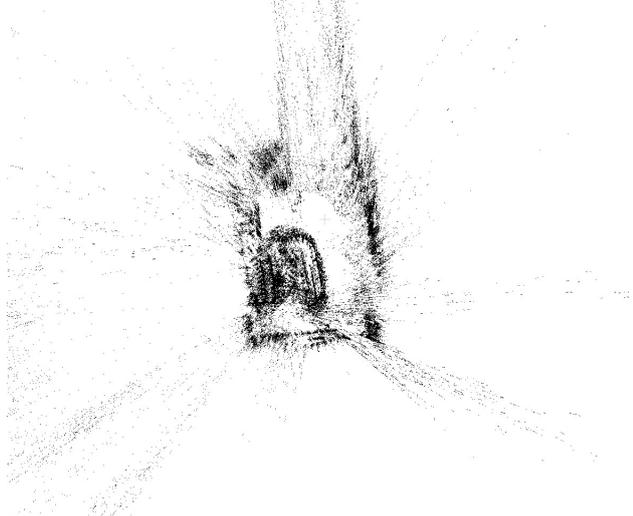


Figure 8. Raw feature point cloud at location A (room 10.O.02)

The resulting coordinate measurement accuracies of the two approaches – feature point-based and planar surface-based – are shown in Table 1. The table lists the average residuals between measured point coordinates and GCP coordinates. For both approaches, the same seven points were measured.

Approach	X [cm]	Y [cm]	Z [cm]	3D [cm]
Point cloud	4	6	5	8.8
Surfaces	2	2	4	4.9

Table 1. Differences between measurement approaches (avg. residuals between measured points and GCP coordinates)

These first results show excellent local point measurement accuracies at the sub-decimetre level for both approaches. They also indicate that the surface-based approach improves the horizontal measuring accuracy by about a factor of two, to approx. 2 cm in X and Y. The vertical point measurement

accuracy does not significantly differ between the two approaches.

Since the planar surface-based approach yields better results and since its handling is simpler, we used this measuring approach for all subsequent investigations.

## 5.2 Trajectories

The accuracy of the trajectories was assessed by comparing the measured check point coordinates along the mapping paths with their reference coordinates. Table 2 lists the RMSE values of for the open (A-C) and the closed loop trajectories (A-A). The RMSE in both horizontal and vertical direction are surprisingly small considering the travelled distance of 90 meters and 140 meters respectively. In the first campaign, the maximum horizontal error is 1.6 meters after 86 meters and the maximum vertical error amounts 0.8 meters after 89 meters along the trajectory (see Figure 10, left). In the second campaign, the maximum horizontal error is 1.7 meters at 104 meters and the maximum vertical 0.4 m after 87 meters along the trajectory (see Figure 10, right).

Trajectory	RMSE GCP			RMSE overall (GCP + Checkpoints)		
	X [cm]	Y [cm]	Z [cm]	X [cm]	Y [cm]	Z [cm]
A – C	4	11	4	55	81	45
A – A	2	2	4	88	35	23

Table 2. Residuals of the one way and the closed loop trajectory

As can be seen in Figure 9, both horizontal and vertical drifts increase with the distance travelled from the start location.

Interestingly, in the closed-loop trajectory at the last measured check point again a high accuracy was proven (<10 cm), as visible in Figure 10. Since the last point was again very close to the location, where the AR application has been initialized, this indicates that the AR device has recognized this location from before and has successfully been relocated to this position. It is important to be aware of this behaviour, since in the current version of the AR framework a) does not signal a loop closure and b) yields a discontinuous trajectory around the loop closure event.

Another interesting phenomenon, which is evident in Figure 9 (left) and especially in Figure 10 (right) is the vertical shift, which happens in the long corridor after around 40 meters. This shift did not happen in the second campaign.

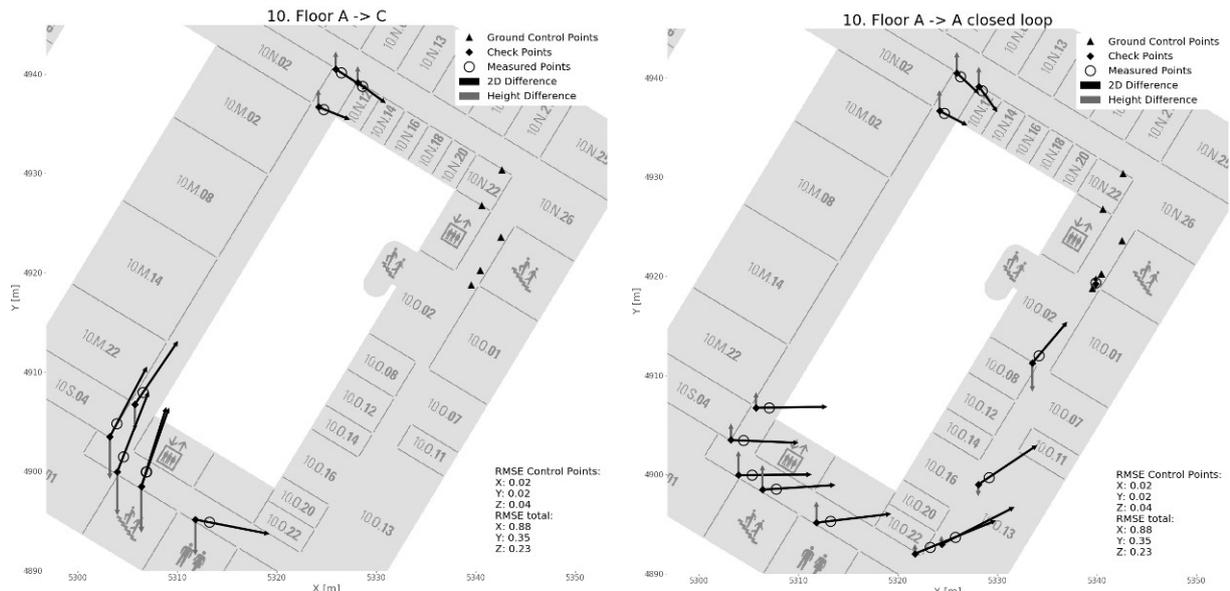


Figure 9. Behaviour of difference vectors (scaled by factor 5) on one-way trajectory (left) and closed-loop trajectory (right)

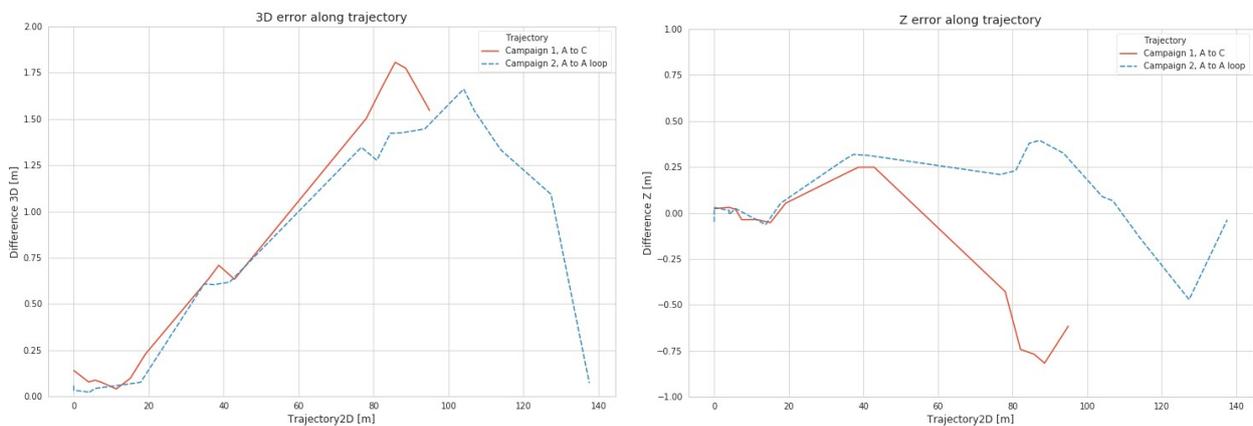


Figure 10. 3D error (left) and Z error (right) along trajectory

## 6. CONCLUSION AND OUTLOOK

We successfully developed a low-cost image-based indoor mobile mapping prototype application based on current AR frameworks. The application runs on most modern mobile devices, both in the iOS and Android ecosystems. Our application supports absolute geo-referencing via ground control points. Thanks to AR tracking, directly georeferenced images can be captured with our application.

In addition, we carried out performance investigations in a challenging indoor environment with two different campaigns. In a first investigation, we obtained local 3D measurements accuracies using the plane-based measuring approach of 2-3 cm horizontally and approx. 5 cm vertically.

Our subsequent mapping test campaigns showed that AR tools are surprisingly accurate with a max 3D error of the full circle campaign of 1.6 m or 1.6% over a distance of 100 meters in a very demanding environment (Figure 10 left). The analysis of the difference vectors in Figure 9 indicates that the local accuracy is even higher. All this shows that AR tools have a huge potential in accurately tracking mobile devices in indoor environments without specific and expensive hardware.

In summary, we demonstrated that AR Frameworks are an interesting alternative to costly high-end mobile mapping systems in certain application areas.

In the future, AR mapping apps could provide a low-cost front-end to an ecosystem of image-based mobile mapping and visual localization services. As demonstrated in this paper, consumer devices could be used for carrying out relatively accurate 3D measurements and for updating existing image-based infrastructure services, e.g. by providing accurately georeferenced error or change reports to facility managers.

Future work includes the combination of the high local accuracy of an AR tool with GNSS or a visual positioning service as an absolute positioning system. We also plan to extend our accuracy investigations to natural outdoor environments without manmade structures.

## ACKNOWLEDGEMENTS

The applied research project cloudIO (No. 32411.1 IP-ICT) was supported by Innosuisse, the Swiss Innovation Agency and by our industry partner iNovitas AG.

## REFERENCES

- Apple Developers. (2019). Augmented Reality - Apple Developer. Retrieved October 16, 2019, from <https://developer.apple.com/augmented-reality/>
- Blaser, S., Cavegn, S., & Nebiker, S. (2018). DEVELOPMENT OF A PORTABLE HIGH PERFORMANCE MOBILE MAPPING SYSTEM USING THE ROBOT OPERATING SYSTEM. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4(1), 13–20. <https://doi.org/10.5194/isprs-annals-IV-1-13-2018>
- Blaser, S., Nebiker, S., & Cavegn, S. (2017). SYSTEM DESIGN, CALIBRATION AND PERFORMANCE ANALYSIS OF A NOVEL 360° STEREO PANORAMIC MOBILE MAPPING SYSTEM. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-1/W1, 207–213. <https://doi.org/10.5194/isprs-annals-IV-1-W1-207-2017>
- Bosse, M., Zlot, R., & Flick, P. (2012). Zebedee: Design of a spring-mounted 3-D range sensor with application to mobile mapping. *IEEE Transactions on Robotics*, 28(5), 1104–1119. <https://doi.org/10.1109/TRO.2012.2200990>
- Google. (n.d.). ARCore - Google Developers | Google Developers. Retrieved October 16, 2019, from <https://developers.google.com/ar>
- Lauterbach, H. A., Borrmann, D., Heß, R., Eck, D., Schilling, K., & Nüchter, A. (2015). Evaluation of a backpack-mounted 3D mobile scanning system. *Remote Sensing*, 7(10), 13753–13781. <https://doi.org/10.3390/rs71013753>
- Lehtola, V. V., Kaartinen, H., Nüchter, A., Kaijaluoto, R., Kukko, A., Litkey, P., ... Hyypää, J. (2017). Comparison of the selected state-of-the-art 3D indoor scanning and point cloud generation methods. *Remote Sensing*, 9(8). <https://doi.org/10.3390/rs9080796>
- Leica Geosystems AG. (2017). *Leica Pegasus: Backpack Mobile reality capture*. Retrieved from <http://leica-geosystems.com/products/mobile-sensor-platforms/capture-platforms/leica-pegasus-backpack>
- Lenherr, P., & Blaser, S. (2018). *3D-Vermessung mit Smartphone Apps Verwendete Applikationen und Smartphone Typen*. 1–8.
- Magicplan. (n.d.). #1 Floor Plan App, Construction & Surveying Software | magicplan. Retrieved October 30, 2019, from <https://www.magicplan.app/en/>
- Miller, D., Mowrer, T., & Weiers, B. (2018). Unity: AR Foundation, ARCore and ARKit. Retrieved October 30, 2019, from <https://blogs.unity3d.com/2018/12/18/unitys-handheld-ar-ecosystem-ar-foundation-arcore-and-arkit/>
- NavVis. (n.d.). NavVis | M6. Retrieved October 28, 2019, from <https://www.navvis.com/m6>
- Nebiker, S., Cavegn, S., & Loesch, B. (2015). Cloud-Based Geospatial 3D Image Spaces—A Powerful Urban Model for the Smart City. *ISPRS International Journal of Geo-Information*, 4(4), 2267–2291. <https://doi.org/10.3390/ijgi4042267>
- Rettenmund, D., Fehr, M., Cavegn, S., & Nebiker, S. (2018). Accurate visual localization in outdoor and indoor environments exploiting 3D image spaces as spatial reference. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 42(1), 355–362. <https://doi.org/10.5194/isprs-archives-XLII-1-355-2018>
- Sarlin, P.-E., Cadena, C., Siegwart, R., & Dymczyk, M. (2019). From Coarse to Fine: Robust Hierarchical Localization at Large Scale. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 12716–12725. Retrieved from <http://arxiv.org/abs/1812.03506>
- Sattler, T., Maddern, W., Toft, C., Torii, A., Hammarstrand, L., Stenborg, E., ... Pajdla, T. (2018). Benchmarking 6DOF Outdoor Visual Localization in Changing Conditions. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 8601–8610. <https://doi.org/10.1109/CVPR.2018.00897>
- Umeyama, S. (1991). Least-Squares Estimation of Transformation Parameters Between Two Point Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4), 376–380. <https://doi.org/10.1109/34.88573>
- Unity. (2018). About AR Foundation | Package Manager UI website. Retrieved October 16, 2019, from Unity 3D website: <https://docs.unity3d.com/Packages/com.unity.xr.foundation@2.2/manual/index.html>
- Widya, A. R., Torii, A., & Okutomi, M. (2018). Structure-from-Motion using Dense CNN Features with Keypoint Relocalization. *IPSJ Transactions on Computer Vision and Applications*, 10(1).
- Zhang, J., & Singh, S. (2017). Low-drift and real-time lidar odometry and mapping. *Autonomous Robots*, 41(2), 401–416. <https://doi.org/10.1007/s10514-016-9548-2>