

GeoAR: a calibration method for Geographic-Aware Augmented Reality

Marcelo L. Galvão, Paolo Fogliaroni, Ioannis Giannopoulos, Gerhard Navratil, Markus Kattenbeck & Negar Alinaghi

To cite this article: Marcelo L. Galvão, Paolo Fogliaroni, Ioannis Giannopoulos, Gerhard Navratil, Markus Kattenbeck & Negar Alinaghi (20 May 2024): GeoAR: a calibration method for Geographic-Aware Augmented Reality, International Journal of Geographical Information Science, DOI: [10.1080/13658816.2024.2355326](https://doi.org/10.1080/13658816.2024.2355326)

To link to this article: <https://doi.org/10.1080/13658816.2024.2355326>



© 2024 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.



Published online: 20 May 2024.



Submit your article to this journal [↗](#)









View related articles [↗](#)



View Crossmark data [↗](#)

GeoAR: a calibration method for Geographic-Aware Augmented Reality

Marcelo L. Galvão , Paolo Fogliaroni , Ioannis Giannopoulos ,
Gerhard Navratil , Markus Kattenbeck  and Negar Alinaghi 

Department of Geodesy and Geoinformation, TU Wien, Vienna, Austria

ABSTRACT

The applicability of augmented reality (AR) as a tool for geospatial data analysis and urban environment interaction relies on developing robust and accurate systems capable of aligning the virtual reference frame with the geographic one. In this article, we introduce our work toward the conceptualization and realization of Geographic-Aware Augmented Reality (GeoAR), including an evaluated framework for the automatic registration of georeferenced AR content. The proposed framework uses a novel calibration method that enables highly accurate placement of augmentations at their assigned geographic coordinate. Moreover, it introduces four calibration approaches suitable for different user needs. The framework was evaluated to assess the robustness of the augmentation's positional accuracy in three areas with different environmental characteristics, using references up to 50 m away while the user moves around. The results demonstrate that this framework supports novel outdoor AR applications, extending the possibilities in research and urban applications.

ARTICLE HISTORY

Received 24 August 2023
Accepted 10 May 2024

KEYWORDS

Augmented reality;
geovisualization; mixed-
reality; geographic
information systems

1. Introduction

Augmented reality (AR) has found its way into indoor spaces, and several works have been published showcasing various scenarios for research and applications. Users can interact with the AR digital content to orient and place them at the desired locations (Hedley *et al.* 2002, Hilliges *et al.* 2012). Moreover, AR applications can use cloud services to store this location relative to the scanned environment to make placement persistent across sessions and devices. Nevertheless, applying AR in the real geographic world, where three-dimensional georeferenced objects must be visualized precisely at their assigned geographic coordinates, remains a challenge (Hansen *et al.* 2021). The process of aligning virtual objects or augmentations with the physical environment is often called registration (Sahu *et al.* 2021); Compared to indoor spaces, registration

CONTACT Ioannis Giannopoulos  igiannopoulos@geo.tuwien.ac.at

© 2024 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.

This is an Open Access article distributed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives License (<http://creativecommons.org/licenses/by-nc-nd/4.0/>), which permits non-commercial re-use, distribution, and reproduction in any medium, provided the original work is properly cited, and is not altered, transformed, or built upon in any way. The terms on which this article has been published allow the posting of the Accepted Manuscript in a repository by the author(s) or with their consent.

outdoors is more challenging due to the large distances and the heterogeneity of the environment (Imottesjo and Kain 2018).

In this article, we present a comprehensive framework that enables an AR system to recognize and localize itself in a geographic space accurately while the user moves around, providing the means to display georeferenced digital content at its exact location. In order to address this problem, for which we suggest the term *making AR geographic-aware*, we propose a method that does not require manual adjustment or accurate compass data. Its novelty consists of collecting control points with redundant GNSS measurements to calculate a 7-parameter Helmert transformation. We provide empirical evidence for the accuracy and robustness of our method by evaluating our proposed system architecture for a SLAM-based Augmented Reality device combined with a high-precision Global Navigation Satellite Systems (GNSS) receiver. Furthermore, we introduce four different calibration approaches using our method that can be useful to different developer needs, including the use of cloud spatial anchors that allow reusing a successful calibration over multiple sessions and devices.

The remainder of this article is structured as follows: [Section 2](#) reviews the literature related to the registration of georeferenced objects. [Section 3](#) introduces the notion of Geographic-Aware AR followed by an open-access framework that describes the technical realization. [Section 4](#) presents the evaluation and results of this framework concerning positional and angular accuracy in three different environmental characteristics; and [Sections 5](#) closes with a discussion about accuracy and applications.

2. Related work

In this section, we provide a literature review on indoor and outdoor AR related to the registration of georeferenced objects, highlighting their differences in terms of calibration method, evaluation, and limitations. Whether indoors or outdoors, AR technology strongly relies on mechanisms to localize the device's relative position to objects in its surroundings. This capability of the AR device to understand the surrounding space, also called spatial awareness by researchers and API providers (Muhammad Nizam *et al.* 2018, Huo *et al.* 2018), is crucial for creating the illusion that the AR content is integrated with the physical objects in the user surroundings. The most common technique for localization uses visual simultaneous localization and mapping (v-SLAM), which consists of continuously scanning the surroundings using the device's cameras and depth sensors. The 3D structures can be mapped and tracked relative to the viewer's pose based on the analysis of scanned objects' from different perspectives. With this information, the AR device is able to calculate the correct perspective of the digital objects to be presented in the device's 2D display to create the augmentation experience. In order to improve the localization accuracy during locomotion, the device's inertial measurement unit (IMU) data, which includes sensors such as accelerometer, gyroscope, and compass, can be processed together with v-SLAM (Low and Lee 2014), also called vi-SLAM. Examples of toolkits using vi-SLAM are Apple's ARKit, Google's ARCore, and Microsoft's Mixed-Reality Toolkit (MRTK). Objects with clearly defined edges, corners, and flat faces, such as walls, columns, or tables, make optimal reference points in the v-SLAM mechanisms. As we discuss in [Section 2](#), such

references are more common and easily detectable in indoor spaces, making v-SLAM less robust and more challenging in outdoor spaces where the scale and distances are larger. A survey on AR SLAM algorithms can be found in Taketomi *et al.* (2017). For simplification, we refer to v-SLAM or vi-SLAM just as SLAM in this article.

Modern approaches addressing the challenge of registering georeferenced objects are commonly divided into: (i) vision-based, which relies on identifying objects in the video streams of the device camera; (ii) sensor-based, which relies on positioning and spatial orientation sensors such as accelerometers, gyroscopes, compasses, magnetometers, or GNSS; and (iii) hybrid solutions, which combine mechanisms from both vision-based and sensor-based (Wu *et al.* 2021, Qiao *et al.* 2019).

Blut and Blankenbach (2021) addressed the challenge of registering georeferenced 3D models for indoor applications using a vision-based approach. In their proof-of-concept, the camera pose in the geographic frame is estimated by extracting the corners of doors from the camera's images and matching them with the corresponding CityGML door model accessed from a database. The authors reported an error of 17 cm in the pose estimation, using doors at 3 and 9 m distance from the camera. The advantage of this vision-based approach is that it supports the AR device in localizing in the geographic frame even without a positioning system. The drawbacks are: first, it depends on external data storage with accurate data models for the calibration; second, the authors did not clarify how the matching of the door with the correct correspondent model can be done automatically; third, as we discuss further, the extraction and matching algorithms are not reliable in all conditions.

Next to the indoor AR challenges, outdoor AR challenges include sunlight glaring visibility of digital content (Wang *et al.* 2017), user safety (Rovira *et al.* 2020), distance perception, and occlusion (Zollmann *et al.* 2021), amongst others. Nevertheless, a primary challenge posed by outdoor AR relates to the robust and precise localization and registration over large areas (Imottesjo and Kain 2018). Compared to indoor spaces, outdoor spaces are more heterogeneous, and the scale and distances are larger. Moreover, reference points that can be properly scanned during the SLAM process are only sometimes present or are too far away (Stachniss *et al.* 2016).

Vision-based approaches in outdoor spaces require powerful object detection and image-matching algorithms to detect physical objects at any distance, scale, and perspective. Such algorithms have high costs that can affect real-time application performance. Additionally, environmental conditions such as light, shadows, or fog affect the algorithm, harming the registration's robustness (Rao *et al.* 2017). Recent research addressed the challenge of improving vision-based registration for outdoor and built environments. Rao *et al.* (2017) explored using deep learning for object detection to improve registration by making it more robust to complex outdoor conditions that involve partial occlusion, motion blur, and perspective variations; however, no evaluation regarding the accuracy was provided. Wu *et al.* (2021) combined the SURF-FREAK (used for object recognition) and KLT (used for matching objects) algorithms to improve the registration performance in real-time applications. Although the authors claim satisfactory accuracy for the registration of buildings, they also did not provide quantitative information from an evaluation in this regard. Despite all recent improvements in the accuracy and robustness of vision-based

approaches, a main drawback in this modality is that it is only possible to register objects in the camera's view range.

Arth *et al.* (2015) addressed this challenge of geo-localization with a hybrid method. Their solution consists of analyzing a video frame and using object detection to identify buildings and match them with 2.5D building data. The technique takes advantage of the usual building's vertical and horizontal lines to first estimate the camera's orientation (rotation). Using additional GNSS data, the x , y , and z translations are estimated in order to achieve global localization. The accuracy of this method depends on how accurately the edges of the building used for the calibration can be extracted. As a consequence, environmental conditions, elements obstructing building parts, and the complexity of the building morphology alike can affect registration accuracy. An evaluation using 32 different buildings showed a camera pose inaccuracy of more than 2 degrees in 15% of all cases. The method we introduce in this paper does neither require object detection algorithms nor external geodatabases for global localization.

The sensor-based approach's core concept is to use the device's geographic position and orientation as a reference to register the AR content. Compared to vision-based, the sensor-based mechanism demands less computation power, and it is potentially more robust because it does not depend on object detection algorithms (Huang *et al.* 2016). Nevertheless, the inaccuracy of the positioning system affects directly registration accuracy. Since AR devices lack accurate geographic positioning systems, sensor-based AR methods are usually applied in outdoor spaces when precise registration is not particularly important (Billinghurst *et al.* 2015, Wu *et al.* 2021, Lee *et al.* 2012). This is the case for many AR navigation and wayfinding approaches (Cron *et al.* 2019, Lee *et al.* 2022). Typically, they are limited to visualizing 3D objects such as arrows or icons in front of the user, i.e., using the device itself as a frame of reference or toward a cardinal direction (Qiu *et al.* 2023, Singh *et al.* 2021). CityViewAR (Lee *et al.* 2012) is another registration method example that uses the device's positioning and orientation to transform the input geographic coordinates relative to the user's position. Huang *et al.* (2016) proposed an interactive sensor-based method to overcome the precise registration of georeferenced objects problem. The proposed method combines real-time kinematic (RTK) corrected GNSS and IMU data for AR localization but no v-SLAM. The registration occurs in two steps. First, the objects' virtual coordinates are estimated using the observer's positioning and orientation. This initial registration roughly matches the virtual object to the corresponding physical one. In order to fine-tune the registration, the user must use a console controller to rotate and translate the virtual object to make the perfect alignment. Burkard and Fuchs-Kittowski (2020) proposed a similar interactive approach aimed at mobile devices that do not require a controller. The manual adjustment works through touchscreen gestures to translate and rotate the 3D models to match the corresponding physical object. This manual matching can be used to localize the AR device in the geographic frame if the 3D models are georeferenced. The method we introduce in this article aims at accurate and robust registration without user manual adjustment but with a location-based calibration.

Our method closely relates to the approach presented by Hansen *et al.* (2021) because it is strictly sensor-based and aimed at semi-automatic accurate registration in large-scale outdoor spaces. Hansen and colleagues designed a self-contained sensor box

with a GNSS-RTK sensor, IMU, and altimetry to provide the AR device (tablet) with accurate geographic coordinates and the north-aligned direction (azimuth). The transformation of the georeferenced objects to the local frame (virtual coordinates) is calculated similarly to the aforementioned related work (Lee *et al.* 2022, Cron *et al.* 2019, Qiu *et al.* 2023, Singh *et al.* 2021, Lee *et al.* 2012, Huang *et al.* 2016), i.e., using a single tuple of geographical position/azimuth. As this method requires azimuthal data for the localization in the global frame, a meticulous IMU-to-camera calibration is necessary, including processing on an external server. The method's accuracy was assessed by comparing concentric circular virtual marks (radius 1, 2, and 3 m) around a physical geodesically verified reference. The error varied in mean with fixed RTK from 3.5 to 4.3 cm for an initial user distance of 4 m. The authors did not provide quantitative information on the degradation or angular accuracy. Our approach does not rely on compass (north-aligned azimuthal data) nor gravitational vectors, which are delicate and difficult to obtain with the required accuracy (e.g., Hansen *et al.* (2021) reported compass error falling within 1–5°). Instead, our method uses several 3D positional records (provided by the GNSS receiver) to estimate the transformation parameters. Moreover, we provide an evaluation with modern survey techniques using references up to 50 m away.

3. Geographic-Aware Augmented Reality

We define *Geographic-Aware AR (GeoAR)* as consisting of AR applications able to align the geographic reference frame with the virtual AR reference frame and keep this alignment while the user moves through the physical world. We use the term GeoAR to clearly define and limit the problem we are addressing, which makes it necessary to differentiate from other terms, such as *AR for Geographic Visualization*, which can be used to visualize geographic data as AR content indoors (Hedley *et al.* 2002); or as discussed in Section 2 the concept of *Outdoor AR* which is often assisted reality applications (Rauschnabel *et al.* 2022) designed for outdoor environments. By analogy to the commonly used term spatial-awareness, geographic-awareness can be defined as the capability of the AR device to recognize and localize itself in the geographic space. Therefore, in contrast to applications that anchor content relative to the user or the surrounding environment scanned in the SLAM process, providing only clues of distance and direction of particular geographic features, GeoAR must be able to accurately display digital content at their assigned 3D geographic coordinates and be robust in preserving their location while the user moves through the environment.

Another important capability of GeoAR is to make the augmentations be perceived as natural compositions with the physical world (Burkard and Fuchs-Kittowski 2020, Postert *et al.* 2021). Despite displaying AR content at its accurate geographic coordinate, the proper user perception of this content's localization in large-scale outdoor spaces is even more challenging than in indoor spaces. In indoor spaces, physical objects can be scanned in the SLAM and modeled into the virtual scene to enable the occlusion effect of AR content by those physical objects. In outdoor space, such scanning is still impractical. However, a precise GeoAR can support AR content to be visualized, respecting occlusion with the physical objects at larger distances (Postert *et al.* 2021). Figure 1 shows a real example of how our GeoAR method can be used to



Figure 1. We use our GeoAR method to highlight landmarks from a panoramic view. By registering a building at its exact location, an AR object behind it can be partly occluded, improving its composition with the physical world. (a) A standard panoramic view with two visible church towers located 300 and 1400 meters away. (b) The churches are augmented with a white rectangular cuboid. Still, the cuboid augmentations do not look naturally arrayed with the physical world because they look like they are floating in front of the buildings. (c) The roof model of some buildings –for which data was collected with LIDAR and represented in CityGML– is added to the AR scene, occluding the cuboids respecting their natural superposition. (d) The building’s roof models get a special shader that is transparent but still hides objects behind them (Postert *et al.* 2021).

create this effect, making the composition with the physical work more natural and improving the distance perception. The precise registration of the building with this special shader make the cuboids be partially hidden behind buildings as they naturally would.

3.1. Achieving geographic awareness

Geographic coordinate systems (GCS) are used to measure and express the position of geographic features or physical objects on the Earth’s surface. GCS, as shown in [Figure 2\(a\)](#), uses an ellipsoidal model of the Earth and a latitude/longitude tuple measured in degrees to map the Earth’s surface; a third coordinate with length unit can be used to measure height, elevation, or altitude. In contrast to GCS, the AR application’s local reference frame is a 3D Cartesian system, as shown in [Figure 2\(b\)](#), which we refer to as the virtual coordinate system (VCS). In order to make AR applications geographically aware, the two coordinate systems need to be aligned. However, a direct spatial transformation between these coordinate systems has no optimal numerical properties and can introduce significant errors. Therefore, the georeferenced objects’ coordinates must first be converted to a projected coordinate system (PCS). This kind of conversion, also known as map projection, is a typical process in cartography to represent the Earth’s curved surface in a Cartesian plane. As map projections inherently imply some distortion, it is important to use an adequate projection for the use area to minimize scale factor inaccuracies (more details in [Section 5.1](#)). Using a PCS serving as the global reference frame makes it possible to estimate the parameters to transform coordinates from the VCS (local reference frame) to the PCS and vice versa. In other words, it is possible to determine to which extent one reference system needs to rotate, translate, and scale to align with the other. The mechanism to find optimal parameters for this transformation is the central process to achieve GeoAR.

Spatial transformation methods between a source and a target Cartesian reference system use control points to estimate the optimal parameters. Control points are points where coordinates are known in both the source and target coordinate

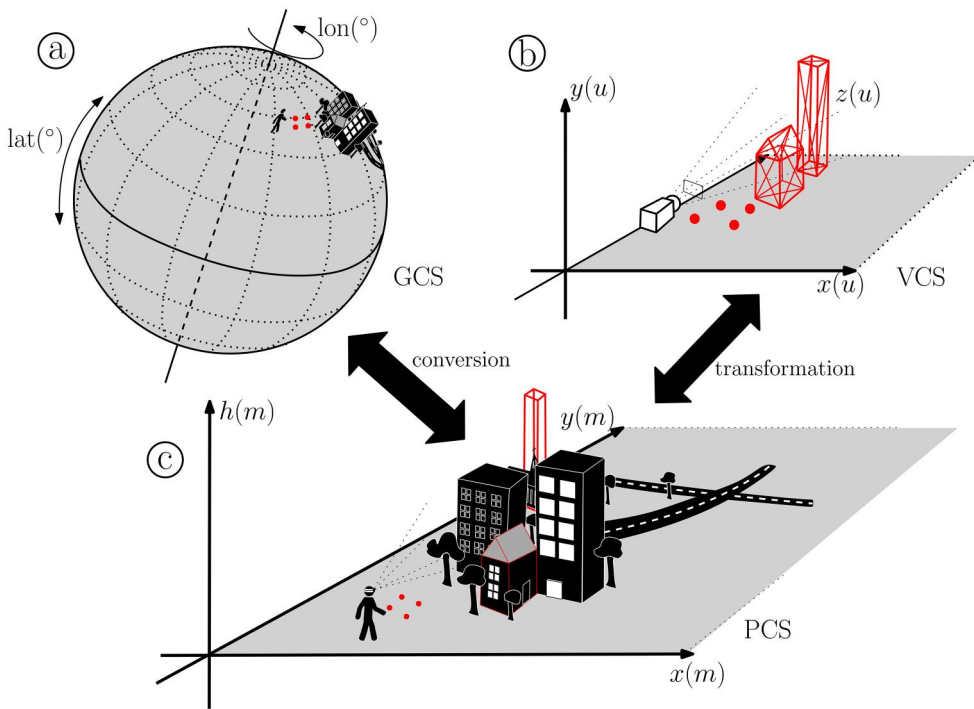


Figure 2. Our method to achieve Geographic-Aware AR consists of collecting correspondent pairs of points in the geographic coordinate systems (a) and in the virtual coordinate system (b). We refer to the pair of points as *control points* (red dots), for which the geographic coordinates are converted to a project coordinate system (c) before estimating the parameters that align the virtual and the geographic frame.

systems. This means it is necessary to match points in the PCS with the VCS; the four red dots in [Figure 2\(b\)](#) represent control points. These are points in the VCS associated with a geographic coordinate, including height. Once coordinates in the GCS and the VCS are obtained for the control points, the transformation parameters can be calculated using the PCS as the source and the virtual as the target. Since both systems have orthogonal axes and constant scale along them, a conformal transformation (single scale factor to all three axes) is sufficient. In such transformations, the parameters are determined as unobserved variables in a system of equations.

Our method uses a seven-parameter transformation, also known as Helmert's transformation method (Ghilani 2010, p. 510 ff), which is a conformal transformation for 3D systems. The seven parameters consist of three translations applied separately along the three axes (x , y , z), three rotations around the three axes, and one scale applied uniformly through the three axes. In order to determine the parameters, two control points and a single coordinate of a third one are required, adding up to a total of seven coordinates. In this case, the system has a single solution because it leads to an equal number of equations and unobserved parameters. Considering that the match of control points is not exact due to inaccuracies (discussed in [Section 5.1](#)), excess control points are used, leading to an over-determined system (more equations than

unobserved parameters). In this case, there is no exact solution, but optimal parameter values can be estimated by least squares adjustment.

Once the transformation parameters are estimated, the coordinates of the georeferenced objects are transformed to the VCS; based on these transformed coordinates, the object can be displayed in the AR user's field of view at its correct position corresponding to the geographic space. The [subsections 3.2](#) and [3.3](#) describe the system components of our GeoAR solution and a calibration process, respectively. Alternative calibration processes to achieve GeoAR are discussed in [Section 5.2](#).

3.2. System architecture

Our proposed architecture is divided into two main execution environments running on a smartphone and a head-mounted AR device ([Figure 3](#)). The third component, the web server, is not mandatory and serves as an alternative to the device's internal storage or to manage the cloud spatial anchors (see [Section 5.2](#)). We implemented and tested the application using an Android smartphone and Microsoft HoloLens. Nevertheless, this architecture can be generalized to different systems as long as they meet two minimal requirements: Most importantly, the AR device should support a robust SLAM method. Second, the AR device needs to support interaction with digital content for the fixed-points calibration method (explained in [Section 5.2](#)). We present the architecture in a generic way; details on the materials and equipment for the evaluations are given in [Section 4.3](#).

The *smartphone* purpose is to collect the user's geographic coordinates and send this information to the GeoAR application through HTTP services published by the self-developed *Geo4HoloLens* application. In order to access high-precision coordinates, the smartphone reads the location from the *GNSS manager* app, which uses RTK corrections providing up to centimeter-level positioning accuracy. RTK services are usually commercial and with limited availability because it depends on base stations to apply the corrections ([Wang et al. 2016](#)).

The GeoAR application has two main modules: (i) The *Calibration Manager* collects the geographic positions and matches them with the AR device's VCS to estimate the transformation parameters. (ii) The *Visualization Manager* loads the georeferenced

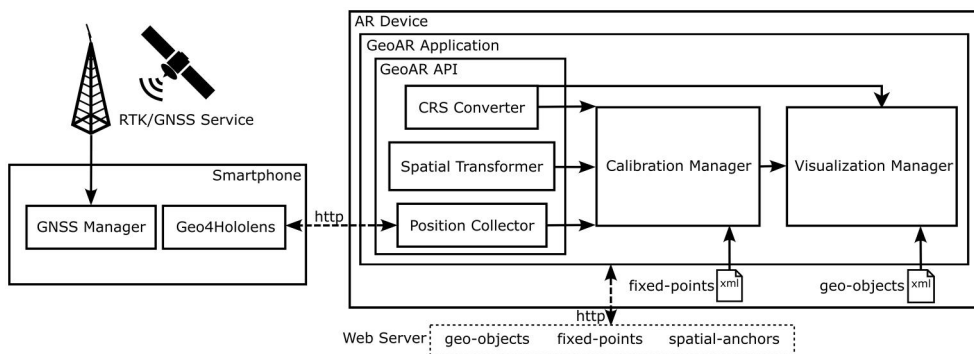


Figure 3. Component structure view of the *GeoAR* framework. The source code and getting started example are available in this research data repository.

objects, transforms their geographic coordinates to the VCS using the parameters provided by the *Calibration Manager*, and renders the objects as AR content. Both modules are supported by the GeoAR API, consisting of three services:

Position Collector: This module supports the *Calibration Manager* by providing the user's geographic position to be used as control points in the calibration process. It requests the user's location (latitude, longitude, and height) via HTTP from the Geo4Hololens service. Due to potential response delays, the request is asynchronous, and the response includes the positioning records' timestamp and measurement accuracy. The records' timestamps are necessary for the calibration process to find their best corresponding point in the VCS.

Spatial Transformer: This module provides two core functions to the *Calibration Manager*. First, it calculates the transformation parameters using the control points provided by the *Calibration Manager*. Second, it provides the estimated parameters to the *Calibration Manager* to support the *Visualization Manager* in making the registration transformation of the geo-objects. Additionally, the *Spatial Transformer* provides methods to validate the transformation parameters by measuring the error (mean residuals) of the transformed control source against the control target points.

CRS Converter: This module provides the methods for the conversion of points or a collection of points from the GCS to the PCS. This service is required by the *Calibration Manager* to convert the collected control points and by the *Visualization Manager* to convert the input geographic coordinates to the projected one.

3.3. GeoAR calibration method

Calibration is the core process in achieving *Geographic-Aware AR* applications, as discussed in [Section 3.1](#). In general, it comprises two main steps: setting the control points and estimating the transformation parameters. In this section, we describe the *location-based* approach for the GeoAR calibration, which is the calibration approach used in the evaluation presented in [Section 4](#). However, our GeoAR calibration can be done with alternative approaches suitable to different user needs. We describe and discuss these alternatives benefits and drawbacks in [Section 5.2](#).

The *location-based calibration* requires an accurate positioning system (GNSS with RTK corrections) but no pre-configuration or setup. This calibration consists of matching the user position with calibration points given by the GeoAR application and then using their coordinates as control points. The calibration starts by showing the calibration points through spherical holograms (calibration spheres) placed at the user's chest height, as shown in [Figure 4\(a\)](#). The calibration spheres are prefabricated game assets associated with an interaction script and owned by the *Calibration Manager*. They are used only to visually indicate the calibration points' location to the user; in fact, the calibration points are transparent cylindrical colliders with a 10 cm diameter and 2 m height. A total of four calibration spheres are rendered, forming a square with a size given by the user.

A user is required to walk into each of these four spheres. Whenever a user walks into a sphere, a sound indicates that the AR device collided with the calibration point.

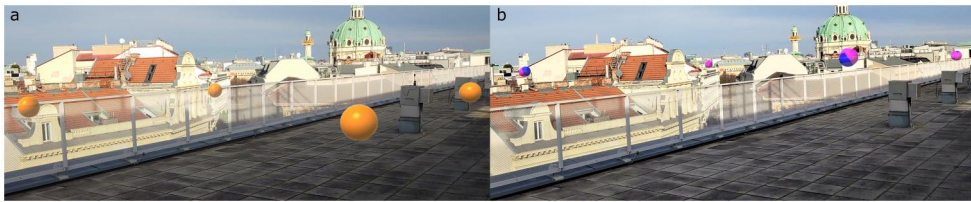


Figure 4. (a) The calibration spheres, in yellow, indicate locations to collect control points. (b) After the calibration, the accuracy spheres can be displayed to validate the calibration; the more the magenta spheres overlap their corresponding blue sphere, the better the control points collection.

From this moment, the *Calibration Manager* requests the user's geographic coordinates (see *Position Collector Section 3*) and records the device's virtual coordinates with their respective timestamps with 2 Hz as long as the device origin keeps touching the calibration sphere's colliders. We call this process points collections, and it is finished when at least three geographic coordinates are collected for each sphere.

The algorithm selects three control points for each calibration sphere. This is done by matching the pair of collected points in the target and source coordinates system with the smallest record time difference. We use three redundant control points for each calibration sphere to reduce the effect of inaccuracies from the positioning system. After the user finishes the points collection for the initial four calibration spheres, the *Calibration Manager* calculates the set of parameters automatically, and the calibration is finished. Figure 4(b) shows a visualization of errors in the calculated parameter from the collected control points. The blue spheres represent the control points' virtual coordinates, and the magenta spheres represent the control points' geographic coordinates after being transformed into virtual coordinates using the estimated parameters.

4. Evaluation

In this section, we report on a series of experiments that estimate the registration accuracy of georeferenced objects with the *GeoAR framework* architecture and procedures. Specifically, we are interested in having a better understanding of (i) the effect of the calibration square size on the registration accuracy and (ii) the performance of the GeoAR components (AR device SLAM and GNSS receivers) in different environmental types. The experiments were designed to evaluate the reliability and usability of the *GeoAR framework* using the location-based calibration method described in Section 3.3. The remaining calibration approaches, described in Section 5.2, were excluded from this evaluation as their distinction lies in the source of coordinates. Fixed-point calibration differs only concerning the origin of coordinates, while for spatial anchors, prior independent calibrations can be loaded, irrespective of the source.

The design of the experiment started from several observations. The first observation is that when leaving the calibration area, the accuracy of the visualization drops, i.e., the offset between real-world and virtual representation will increase. The second observation is that the calibration result was not equally good for the four spheres. This led to the test for directional errors on the loss of positional accuracy. The third

observation is that the positional accuracy varied according to the characteristics of the environment, e.g., open space, street corridors, and green areas. Thus, we addressed the following questions:

1. What is the positional registration accuracy of the *GeoAR framework*?
2. How does moving around and away from the calibration area affect the accuracy?
3. Is the *GeoAR framework* functional across different types of urban environments?

To answer those questions, we measured and compared the visualized location of virtual objects (projected points) with their target location, i.e., the objects assigned geographic coordinates locations (target points). The target and projected points were measured with high-precision measurement tools, i.e., a GNSS receiver supported by RTK, provided by Echtzeit Positionierung Austria (EPOSA) in our case, and a total station, as described in [Section 4.4](#).

To analyze how the registration accuracy varies through the space, we arranged the target points following the structure of a polar grid (see [section 4.1.1](#)). The grid was designed in a pattern that allows us to study how the registration accuracy varies as target points revolve around and get away from the calibration area. To assess the reliability of the *GeoAR framework* across different types of urban environments, we perform the accuracy measurements at three locations with different characteristics (see [section 4.2](#)) that challenge different components of the *GeoAR framework*.

4.1. Experimental design

The two main dependent (measured) variables are the projected points registration *positional accuracy* and *angular accuracy*. The independent variables are: (i) the polar coordinates (θ, r) of the target point with respect to the calibration center and organized as the polar grid model, (ii) the *calibration square size* (5 and 10 m), and (iii) the *environmental types* (urban open space, urban canyon, and green area).

4.1.1. Evaluation grid and calibration points

The reference points for the evaluation are arranged as in the polar grid model shown in [Figure 5](#). We placed a reference point labeled p_0 at the center of the calibration area. This point also serves as the origin of the grid frame of reference. To analyze how positional accuracy varies as we move around and away from the calibration area, we designed the grid to include a set of 24 reference points equally distributed on three concentric circles centered in p_0 and of radius (r) equal to 5, 10, and 15 m. The points on each circle are distributed along eight directions that equally divide the revolution and with the i -th direction forming an angle $\theta_i = 45^\circ \cdot i$ with the positive x -axis of the grid reference system. Each such point can be identified by its polar coordinates $p = (\theta, r)$. This configuration of the evaluation grid will provide insights into how the accuracy varies with the radial distance from p_0 and the angle θ .

Finally, to inspect the performance at greater distances from the calibration area, we included seven more points along the positive x -axis of the grid frame of

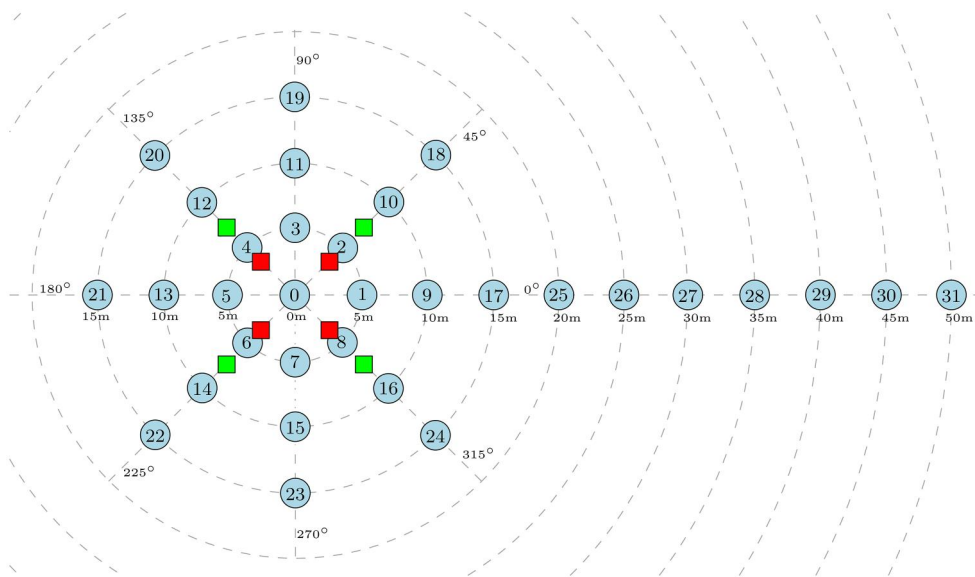


Figure 5. Polar grid model for the target reference points. The target reference points are distributed in eight directions and with increasing radial distance from the origin of the grid reference system p_0 . We included more points along one axis to collect observations at greater distances from the calibration area. The red and green squares represent the calibration points for 5×5 m and 10×10 m, respectively.

reference, equally spaced 5 m apart and reaching a maximum distance of 50 m from p_0 . We extended in a single direction to facilitate finding areas that would fit the grid.

In our experiments, we calibrated using the *GeoAR framework*, collecting calibration points located at the corners of a square centered in p_0 and with the edges parallel to the axes of the reference system of the evaluation grid. We performed experiments with two calibration areas of sides 5 and 10 m. The two different sets of calibration points are also reported in [Figure 5](#) for comparison.

4.2. Tested environmental types

Considering that the diverse characteristics of different environments can affect the *GeoAR framework* robustness and accuracy, we conducted the evaluation in three different types of environments within a city.

4.2.1. Baseline urban space

This test environment represents urban areas where both localization components of the *GeoAR framework* are known to operate seamlessly, which would allow for optimal calibration. These areas are characterized by good GNSS coverage and many easily identifiable visual features. Such characteristics ensure that calibration points can be acquired with high accuracy in both the geographic (GCS) and virtual coordinate reference systems (VCS). Examples of such environments include squares, large streets, or even narrower streets surrounded by low buildings. For our evaluation, we selected

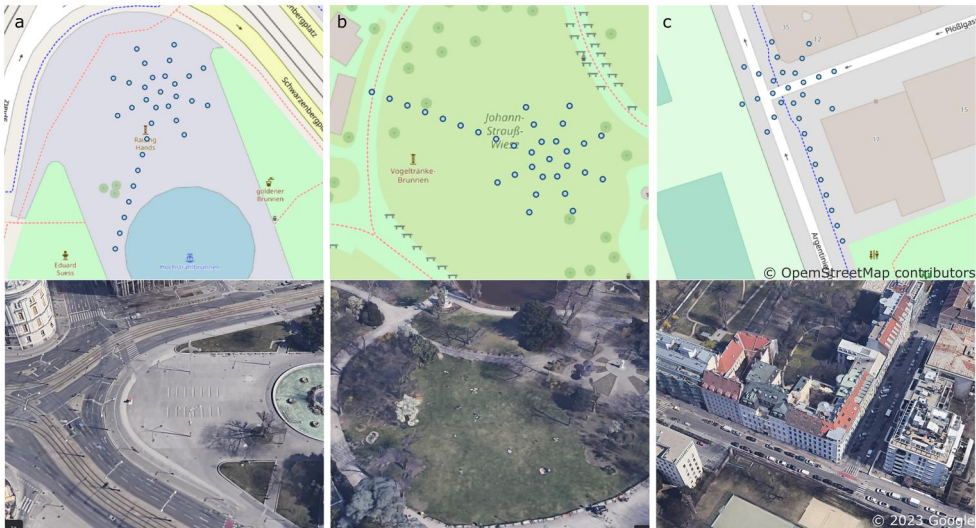


Figure 6. Tested environmental types: (a) Baseline urban space; (b) Green area; (c) Urban canyons (7 observations could not be measured because they were projected inside the building or over a fence).

the square in [Figure 6\(a\)](#). The square is very open, so GNSS reception is optimal throughout the day, and the buildings, a limited number of trees, and a large monumental fountain serve as visual landmarks for the HoloLens' SLAM algorithms. We have carefully chosen the experimental area in a way such that the impact of treetops potentially impacting the GNSS reception is minimized.

4.2.2. Green areas

Our evaluated green area is a typical urban park with grassland and trees, as shown in [Figure 6\(b\)](#). While GNSS reception is optimal throughout the day in such an environment, it is more challenging for SLAM algorithms due to the limited availability of visual landmarks. Forests or green areas with higher tree coverage pose challenges that are not part of the present evaluation.

4.2.3. Urban canyons

Urban canyons are streets surrounded on both sides by tall continuous buildings. In such environments, the street-to-building height ratio is such that only a small portion of the sky is visible at any given time. This has a negative impact on GNSS coverage, as satellites are only visible when passing almost perpendicular to the street. In addition, GNSS positioning in this type of environment is known to suffer from multipath (Hannah 2001): satellite signals bounce off building facades multiple times before reaching the GNSS antenna. Conversely, urban canyons typically feature many easily distinguishable landmarks, providing more references for SLAM. For our evaluation, we selected the street in [Figure 6\(c\)](#).

4.3. Materials

The AR device used in this study was the optical see-through head-mounted display Microsoft HoloLens. The HoloLens is a spatial-aware, wearable computer equipped with an array of sensors. Sensor data is continuously processed by proprietary SLAM algorithms to provide a continuous understanding of the device's position and orientation in the environment. The primary sensors used for the SLAM process are the IMU and four cameras pointed at the front and sides of the device. Position and orientation are expressed in the VCS with origin at the location where the HoloLens is located when the AR application is started. The VCS is a left-handed coordinate system with the x , y , and z axis, respectively aligned with the lateral, vertical, and longitudinal axis of the HoloLens when the AR application is started.

The GNSS receiver used in this study is a PPM 10xx, mounting a Trimble MB-Two board with firmware version 3.62 (total weight of 230 g). The raw GNSS positions computed onboard of the PPM 10xx are streamed through a USB cable to a companion application called PPM Commander running on an Android device (*GNSS Manager* in Figure 3). The PPM commander is registered to the EPOSA RTK correction service. It redirects the received GNSS positions to the RTK service and returns the correction data. It then uses the EPOSA data to correct the GNSS coordinates coming from the receiver. A second application (Geo4HoloLens) publishes the corrected coordinates encoded using the National Marine Electronics Association (NMEA) string format through an HTTP service. Finally, the HoloLens can request the coordinates using the *Position Collector* module.

4.4. Procedure

For the experiment, we developed a HoloLens application using the *GeoAR framework* that reads a csv file with geographic coordinates (including the altitude) and visualizes them as displayed in Figure 7(a). Each point is displayed as a 5-cm sphere at the target coordinate. The sphere is projected on the ground where a disk of the same radius is displayed. A 2-m height white pole (vertical line) extends from the center of the projection, where a white panel displays the point ID and its coordinates.

We conducted the same experiment in each of the three experimental locations described in Section 4.2, repeating twice for each location using

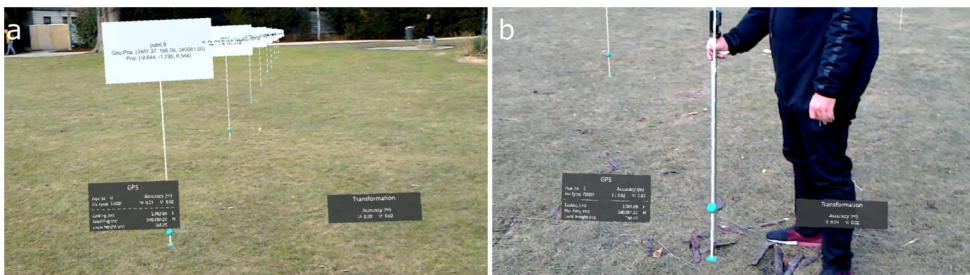


Figure 7. Evaluation of the registration accuracy. (a) After the calibration, the reference points are projected using the estimated parameters and visualized as a 5 cm blue sphere, plus a vertical pole and annotations. (b) The actual positions of the projected points are first marked and later measured using the GNSS antenna with RTK corrections.

calibration sizes of 5×5 m and 10×10 m, as illustrated in [Figure 5](#). The procedure has three phases:

1. *Initial setup*: First, we identified an area large enough to accommodate our polar grid model ([Figure 5](#)) for each experimental type. Then, we marked the point p_0 on the ground and collected its geographic coordinates with the high-precision GNSS receiver described in [Section 4.3](#), ensuring that the RTK correction was functioning and that the horizontal accuracy was within 2 cm. We used a total station centered on p_0 to mark on the ground point p_{31} and the two sets of calibration points, the 5 m and the 10 m. We used the GNSS receiver to collect the geographic coordinates of p_{31} and compute the geographic coordinates of the remaining target points (p_1 – p_{30}). The target geographic coordinates were stored in an internal csv file and used to assess the accuracy of the registration.
2. *Calibration*: For the sake of consistency in experimental conditions, we performed a cache reset of the HoloLens at the beginning of each calibration phase. Indeed, if the cache contained data about the testing environment, the HoloLens would already have an existing understanding of that spatial location, which could affect the accuracy of the positioning. With a cleared cache, we started the AR application and performed a location-based calibration as described in [Section 3.3](#). Making sure the GNSS receiver provides positions with a horizontal accuracy within 2 cm, we moved to the calibration spheres and waited for the control points to be collected. After collecting the control points for all four spheres, the AR application computed the transformation parameters to map from the GCS to the VCS; we used EPSG:31256 as the source PCS for calibration, which is the conformal projection most accurate projection for our experimental area. Finally, the parameters are used to transform the stored target points' geographic coordinates and project each of them as displayed in [Figure 7\(a\)](#).
3. *Measurement*: The measurement phase was conducted with the assistance of a second person. The person using the HoloLens approached the first projection of the first target point and guided the second person to place the tip of a stick at the location where the reference point was projected on the ground, as illustrated in [Figure 7\(b\)](#). This process was repeated from different view angles and orthogonal perspectives to make sure that the stick was located in the right position as the projection. As a final validation, the experimental assistant was asked to set the stick in a vertical position without moving the tip from the ground. If the stick was placed at the exact projected points, there is a perfect match with the projections' virtual pole. When the HoloLens user confirms the match, the second person marks on the ground the position pinned by the stick. The same operation was repeated for the remainder of the reference points following their index order. Once all the points were marked on the ground, we used the GNSS receiver with the RTK correction to collect their geographic locations.

4.5. Results

The experiment generated a total of 185 observations out of the expected 192. Seven observations for the urban canyon environment are missing because they were projected inside a building or over a fence and could not be measured. [Figure 8](#) shows

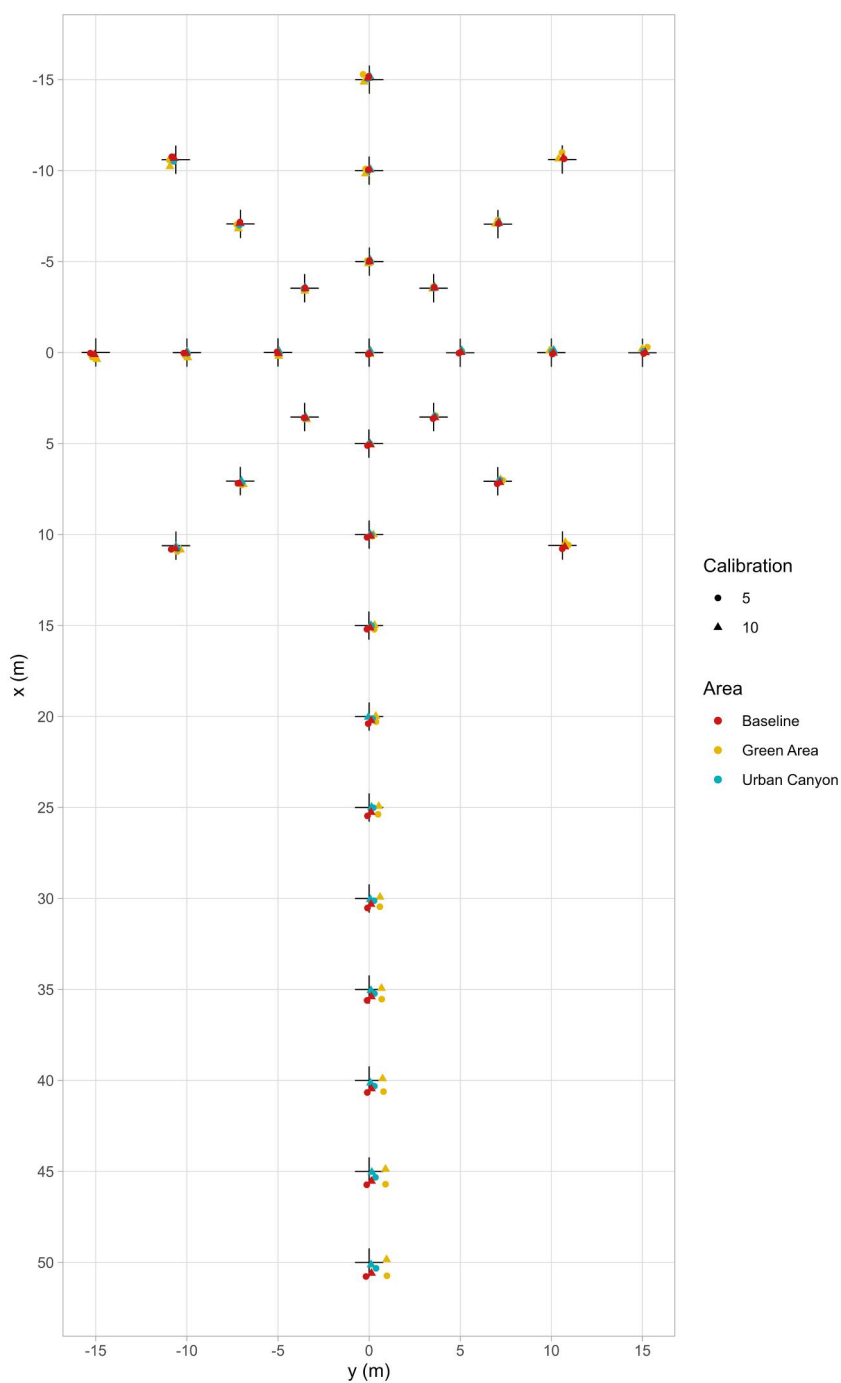


Figure 8. Normalized plot of all observations. Each of the black crosses represents the target position of a reference point. The circle and the triangles represent the measured position for the 5×5 m and 10×10 m calibration, respectively. The environmental types are distinguished by different colors.

the spatial distribution of the 185 observations compared to the target positions. To easily compare them across calibration sizes and environmental types, we normalized the measured coordinates to be centered on the reference point p_0 and rotated them by the angle θ formed by the vector $\overline{p_0 p_{31}}$ with the x-axis. The black crosses represent the target points with known geographic coordinates, and the circles and the triangles are the projected points for the calibration with 5 and 10 meters square size, respectively. Note that approximately up to 10 meters away from p_0 , the projected points are very clustered and centered at their respective target point, making it difficult to distinguish the individual observations because they overlap. However, the further away the observations are from p_0 , the more they spread apart from their respective reference point. This effect was expected, considering the inherent transformation parameter errors. In the next section, we analyze the results to understand this degradation better.

4.5.1. Positional registration accuracy

We separately evaluated the effect of target point distance on registration positional accuracy for calibration size and environment area. The target point distance refers to the distance from the reference point to p_0 , and the positional accuracy is the Euclidean distance between the projected point and its respective target point.

Linear regression was used to test if the target points' distance from p_0 significantly predicted the positional registration accuracy. Table 1 summarizes the results. Overall, the regression was statistically significant, confirming that the distance from the calibration area negatively affects registration accuracy. As expected, the 10 m calibration demonstrated a lower degradation than the 5 m across all environment types. Specifically to the difference across the environmental types, the accuracy in the urban canyon area was the best, and in the green area, the worst. The smallest degradation was 0.2 cm/m in the urban canyon environment with a 10 m calibration, while the highest was 2.6 cm/m in the green area with a 5 m calibration. Figure 9 illustrates the linear model with 95% confidence interval bands (gray band).

4.5.2. Angular registration accuracy

For the angular accuracy evaluation, we calculate the azimuthal difference between the projected points and their respective reference point on the grid with respect to p_0 . A linear regression did not reveal significant differences (the smallest p-value found was $p=0.19$). This result is expected since the angular error should remain constant independent of the distance. In fact, points very close to p_0 presented a higher angular error. This can be explained by the fact that measurement errors have a stronger influence on angular measurements for points closer to the reference origin

Table 1. Analysis of the registration positional accuracy.

Area	Calibration	Degradation (m/m)	Std Dev (m/m)	p-Value	R^2
Baseline	5 × 5	0.017	0.0009	<0.001	0.9182
	10 × 10	0.012	0.0005	<0.001	0.9523
Canyon	5 × 5	0.011	0.0005	<0.001	0.9458
	10 × 10	0.002	0.0006	0.0017	0.3202
Green	5 × 5	0.026	0.0005	<0.001	0.9897
	10 × 10	0.019	0.001	<0.001	0.9237

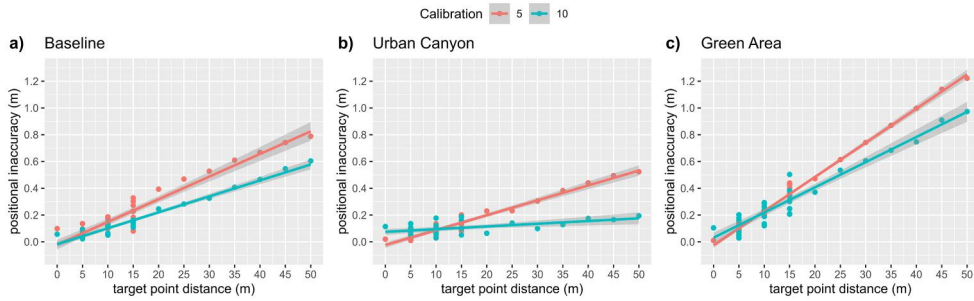


Figure 9. Registration positional accuracy linear regression.

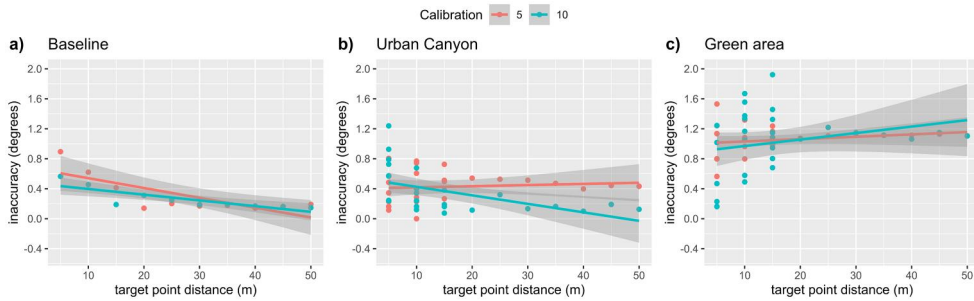


Figure 10. Angular inaccuracy linear regression.

Table 2. Analysis of the registration angular inaccuracy: mean and standard deviation considering only points with distance ≥ 20 m.

Area	Calibration	Angular Inaccuracy ($^{\circ}$)	Std Dev ($^{\circ}$)
Baseline	5 \times 5	0.169	0.023
	10 \times 10	0.204	0.058
Green Area	5 \times 5	1.107	0.016
	10 \times 10	1.122	0.055
Urban Canyon	5 \times 5	0.475	0.053
	10 \times 10	0.164	0.075

p_0 . Figure 10 illustrates the linear model with 95% confidence interval bands (gray band). The inaccuracy (in degrees) converges to a constant in the plots for all three areas. Furthermore, the standard deviation decreases for reference points with distance ≥ 20 m. Therefore, Table 2 provides the means and standard deviations, including only measurements with a distance ≥ 20 m from p_0 . The lowest estimated angle inaccuracy was 0.16° , achieved in the baseline 5 \times 5, and the highest was 1.12° , achieved in the green area 10 \times 10.

5. Discussion

The evaluation of our approach focused on the accuracy of the registration, where we considered the effect of the calibration area and the feasibility for different outdoor environments. We evaluated two related accuracy variables: (i) positional accuracy, which measures the displacement of the virtual object compared to its target real

position; (ii) angular accuracy, which measures the azimuthal difference between the virtual object and its target position with respect to the user. As expected, the larger the calibration area, the higher the positional accuracy. Similarly, the positional accuracy slightly degrades as the user moves further away from the calibration area. Using our location-based calibration method, we found a degradation varying from 0.002 to 0.026 m/m. By way of illustration, the registration of an object 100 m away would represent a displacement of 1.1 m and 2.6 m, respectively. Regarding the angular accuracy, first, we did not identify any degradation, which indicates that the error is mainly derived from the errors in the estimation of rotation parameters. Also, increasing the calibration area from 5×5 m to 10×10 m did not affect the angular accuracy. The estimated angle inaccuracy varied from 0.16° to 1.12° . One limitation of our evaluation is that we did not assess the vertical error; in future work, we intend to extend the evaluation to overcome this limitation.

Comparing the accuracy of our method with previous research addressing the GeoAR problem is challenging because of the diverse nature of solutions and devices. For instance, while Arth *et al.* (2015) assessed the accuracy by measuring the camera pose error in the GCS, Hansen *et al.* (2021) used snap-shots from the mobile AR device to measure the displacement of the virtual markers with respect to georeferenced physical references within 4 m distance. In our evaluation, we are not interested in the camera pose error but the actual augmentations positional error. We also aimed our evaluation to understand the degradation and angular accuracy using further away references while the user moves around. The results provide reliable values on the capabilities of our GeoAR framework in different environments regarding robustness and accuracy, which do not exceed the ones found in literature, even for considerably larger distances and varying environments, highlighting its potential. Future research can investigate if such accuracy represents benefits to applications such as highlighting a destination in an AR-based navigation system Mazurkiewicz *et al.* (2023), or create an occlusion effect to improve depth perception (Figure 1).

Regarding the environmental types, the positional accuracy was best for the urban canyon 10×10 and worst for the green area 5×5 . Although it is not possible to confirm a significant effect, we provide two potential factors affecting registration accuracy. First, the RTK corrections in green areas could be worse than for streets, thus leading to higher errors in the parameters. The second factor could be the SLAM, which depends on detecting objects in the surroundings to improve the device's spatial awareness. As urban canyons have more detectable objects than green areas, this might result in the higher accuracy achieved in the former environment. We further discuss factors affecting accuracy in Section 5.1; nevertheless, future research is needed to shed more light on this question.

5.1. Factors affecting registration accuracy

The precision of the registration is of utmost importance for many aspects regarding functionality, user experience, and perception (e.g., misplacements can impair the interpretation of augmentations). The most critical errors pertain to parameter optimization. The Helmert transformation has seven parameters, so there are seven

variables that impact the calibration, and the further the geo-objects are from the calibration points, the larger the displacements will be. We identify three sources of calibration errors, all of which need to be tackled together because their individual inaccuracies are summed to negatively affect the overall calibration quality.

- *Horizontal and vertical positioning accuracy*: Accurately estimating the calibration parameters is hindered primarily by the challenge of precisely ascertaining the location's longitude, latitude, and altitude. For instance, GPS-enabled smartphones usually provide accuracy within a 3-meter radius, rendering GeoAR impractical for applications that demand centimeter-level precision.
- *GNSS receiver's relative position to the AR device reference origin*: Optimally, device and GNSS receiver positions should align, as the device's virtual coordinates serve as target control points for calibration parameters. The MRTK default origin is the center of the head-mounted device, but this origin position can be shifted to match the actual GNSS receiver (antenna). It is crucial to keep the relative position of the receiver and the HoloLens as constant as possible. Nevertheless, the bending of their neck by users can affect the precision.
- *Calibration spheres colliders*: The location-based calibration uses the calibration spheres to approximate the collected user's geographic location with the user's virtual location. This approximation is detected by colliders of calibration points touching the device's collider. As we collect redundant controls for each calibration sphere to reduce the effect of inaccuracies from the positioning system, the thinner the colliders, the better the approximation. The disadvantage of this is that a certain minimum diameter is required to make it reasonably easy for a user to keep them touching. We use a 5 cm radius cylindrical collider; additionally, we provide sound interaction with calibration spheres indicating the entering and exiting of the colliders.

Beyond the mentioned calibration sources, three other non-calibration-related aspects can adversely affect registration accuracy and should be considered.

- *SLAM robustness*: The registration's robustness relies on the device's spatial awareness during the SLAM process. SLAM approaches continuously scan and update the environment based on sensory input. Changes in the environment representation can impact positional and angular accuracy, causing the virtual CRS origin to shift. Consequently, the augmentations position in the user's field of view may shift despite the actual coordinates remaining constant.
- *Geo-objects coordinates accuracy*: Georeferenced object coordinates inherently have measurement inaccuracies. Calibration will not suffice if assigned coordinates are highly inaccurate. Vertical coordinates, in particular, tend to lack precision, posing challenges in elevation-varying regions like hills or slopes.
- *Scale factor variation in map projections*: As explained in [Section 3.1](#), the registration transformation uses similarity transformation from a PCS to the VCS, both systems with constant scale along the axes and orthogonal. However, map projections of the earth (ellipsoid) always result in a continuous variation in scale. For instance,

the Mercator projection in high latitudes has a high discrepancy in the scale between meridians and parallels. This can cause a significant registration inaccuracy for points outside the calibration area. However, if the PCS has high accuracy for the usage area, this scale variation is neglectable within the range of several kilometers.

5.2. Alternative GeoAR calibration approaches

In our evaluation, we used a location-based approach for our GeoAR calibration (Section 3.3). However, our calibration method can be achieved with different approaches which might be suitable to different developer needs. Therefore, in the interest of making this method as accessible to as many people as possible, we present three alternatives to the *location-based* approach for the GeoAR calibration. Table 3 highlights their key benefits and drawbacks.

5.2.1. Fixed-points calibration

The fixed-points approach can be performed without a GNSS receiver, contrary to the location-based approach. Four points marked on the ground with previously measured coordinates are used instead. The user is asked to grab and place four virtual spheres on the corresponding points. Subsequently, the *Calibration Manager* uses the spheres' virtual position and the marked points' geographic coordinates as control points to calculate the transformation parameters. The coordinates can be measured with a GNSS receiver or by using an existing survey maker as a reference for the fixed points, thus eliminating the necessity for an expensive GNSS receiver. However, the fixed-point coordinates are set for the specific use area and must be accessible by the AR Device, either from the device's internal storage or from a web service. This approach can be less precise than the location-based approach as it depends on how accurately the user can position the virtual spheres at the markers. This method can be considered suitable for a user study where several calibrations need to be made for the same area; for instance, Mazurkiewicz *et al.* (2023) used our fixed-points calibration for their study about AR-assisted navigation.

5.2.2. Persistent calibration through cloud spatial anchors

A calibration performed using our GeoAR application can be persisted across sessions, devices, or applications by storing the control points as the so-called *spatial anchors*. Spatial anchors are an AR concept that enables the persistence of a position in space relative to the SLAM-scanned environment. This is done by storing the scanned environment in cloud services such that the AR application can use this data to match its

Table 3. Calibration methods' features and differences.

Approach	Accuracy	GNSS	Pre-configuration	Pre-calibration
<i>Location-based</i>	Higher	Required	Not required	Not required
<i>Fixed-points</i>	Lower	Not required	Required	Not required
<i>On the fly adjustment</i>	Higher	Required	Not required	Required
<i>Spatial anchors</i>	Higher	Not required	Required	Required

Columns descriptions: (i) method name, (ii) level of registration accuracy, (iii) GNSS receiver is required, (iv) Pre-configuration is required (v) A previous initial calibration is required.

current SLAM-based environment. Our GeoAR application can use this mechanism by generating anchors for the calibration control points stored together with collected geographic coordinates. Starting a different session in the same area, the control points' anchors are reloaded in the same position as they were initially collected, and the transformation parameters can be calculated. Our implementation uses the Microsoft Azure Spatial Anchor Service to achieve this, but our architecture is open to any other spatial anchor approach.

5.2.3. On-the-fly calibration adjustment

The registration inaccuracy increases proportionally to the user's distance from the calibration control points. Therefore, adding control points to the calibration while the user moves away from the initial calibration area can improve the registration. We propose two methods for the on-the-fly calibration adjustment: voluntary and ubiquitous. Both methods require an initial calibration. In the *voluntary adjustment*, the user can always request a new calibration sphere after the initial calibration. In this case, the calibration sphere will be placed just in front of the user. Then, the user performs the point collection as in the location-based calibration, and the transformation parameters are recalculated. The *ubiquitous adjustment* collects control points and recalculates the parameters without user request. The challenge involved in this approach is that during movement, the excellent matching of the control points is affected. Therefore, it is necessary to define quality criteria to collect the control points: (i) The user needs to stop for a minimal period of time; (ii) the positioning accuracy must be equivalent to initial calibration; (iii) maximal virtual coordinate offset; (iv) maximal time offset between the collected points. If the collected control points meet the criteria, the *Calibration Manager* uses them with the previously collected control points to recalculate the transformation parameters. This calibration is suitable when mobility over long distances is required, but more research is required to understand the optimal way to collect the control points on the fly.

5.3. Application scenarios

The introduced GeoAR technology offers a wide range of opportunities in research and application. We introduce some of the most promising scenarios focusing on visualization, sensing, and interaction.

5.3.1. Georeferenced data visualization

The GeoAR technology can be used by various industries to facilitate their business processes. For example, the underground infrastructure of pipelines can be visualized with high accuracy (Figure 11(b)), allowing inspection of the network and maintenance excavations (Figure 11(a)). In the same way, georeferenced objects can be visualized attached to the Earth's surface, such as navigation arrows and excavation markers, or displayed elevated from the ground, such as signs for workers or any kind of indicators for pedestrians (Figure 11(c)).

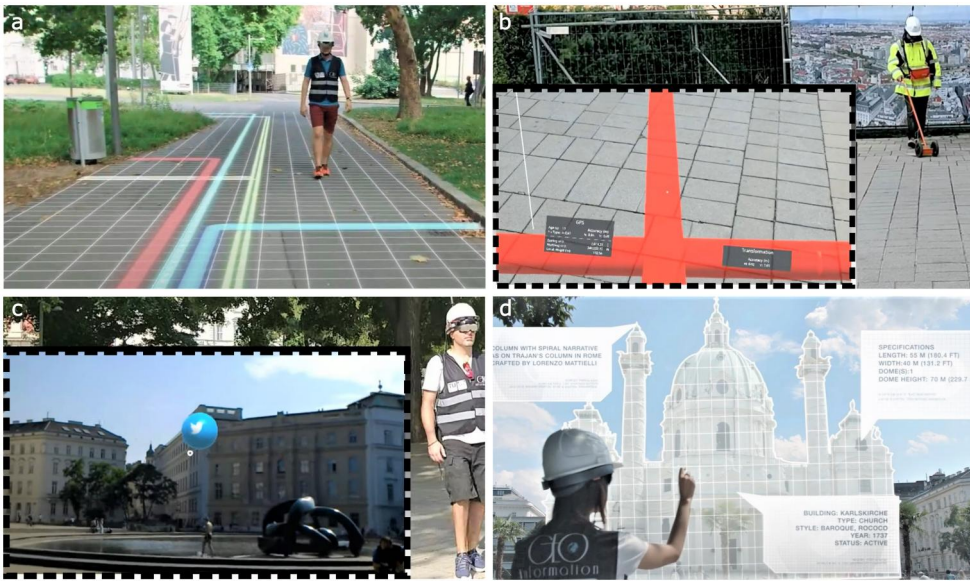


Figure 11. GeoAR application scenarios: (a) Visualization of hidden structures. (b) Underground pipe infrastructure. (c) Geo crowd-sourced data collection. (d) Urban environment engagement and interaction.

5.3.2. Geographic feature interaction

By having a three-dimensional model of the city with information on buildings and infrastructure, the GeoAR technology can be used to make the associated real objects interactive. The process is similar to visualizing georeferenced data, but instead, transparency is used to avoid overlay with the physical environment. Of course, interaction dialogs must also be implemented. For example, one can place a transparent object over a building in the real environment (based on coordinates from the city model) and implement an interaction with air gestures. Once this interaction is triggered, the system can use information from the city model or online resources for assistance, provide information, and collect crowd-sourced data (Imottesjo and Kain 2018) (see Figure 11(d)).

5.3.3. Georeferenced data acquisition

GeoAR applications can be used to collect geographic data and make spatial measurements by reversing the transformation. This way, the scanned objects in the SLAM process can be georeferenced. Moreover, users can obtain geographic coordinates of points of difficult access by using the new interaction capabilities, such as so-called *far interaction* (Lilligreen and Wiebel 2023).

6. Conclusion

Positioning visual content in Outdoor AR applications can either be placed relative to a user's position or using the coordinates of georeferenced features. This paper focuses on solving the problem of making AR geographic-aware, which consists of AR applications capable of aligning the geographic reference frame with the virtual AR

reference frame and keeping this alignment while the user moves through the physical world. We addressed this problem by developing a framework called *Geographic-Aware Augmented Reality* (GeoAR) that seamlessly aligns in a global reference frame with the geographic frame. Our method uses several 3D positional records to estimate the transformation parameters. Moreover, we provide an evaluation to assess the augmentation's positional accuracy using standard surveying technology for references up to 50m away while the user moves around. In contrast to previous research, our results provide reliable quantification of the positional and angular accuracy for registering georeferenced content. Our novel GeoAR calibration represents progress in terms of usability and accessibility because: (i) it achieves accurate and robust registration without manual adjustment by users; (ii) it does not require an external geodatabase for global localization; (iii) it does not rely on compass measurements nor gravitational vectors which are delicate and difficult to obtain with the required accuracy; (iv) it is independent of object detection algorithms, which is costly and often not robust. We expect this work to provide new means of geospatial data analysis and urban environment engagement through various AR applications.

Acknowledgments

We would like to thank Bartosz Mazurkiewicz and David Kirchsteiger for their contributions to the development of this research software. We also acknowledge TU Wien Bibliothek for financial support through its Open Access Funding Programme. Additionally, we appreciate the constructive feedback provided by five anonymous reviewers and the editor on the paper. Parts of this research were funded by Wiener Netze GmbH, which we gratefully acknowledge.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Notes on contributors

Marcelo de Lima Galvão is a computer scientist with PhD in geoinformatics and a postdoc researcher in the Geoinformation Group at TU Wien. He is interested in how novel computer visualization can improve wayfinding and orientation. His CRediT roles for this paper were Writing, Visualization, Conceptualization, Software, Methodology, and Formal analysis.

Paolo Fogliaroni is currently serving as a product owner and the leader of the research team at Esri's Vienna R&D center, focusing on indoor positioning systems. During his postdoctoral stay at TU Wien, he focused on GIS, human-computer interaction, and virtual and augmented reality. His CRediT roles for this paper are Conceptualization, Formal Analysis, Investigation, Methodology, Software, and Writing (original draft).

Ioannis Giannopoulos is a Full Professor for Geoinformation at TU Wien. His research interests lie in Urban Computing, geoAI, Mobile and Remote Eye Tracking, Pedestrian Navigation and Mixed Environments. His CRediT roles for this paper are Conceptualization, Funding acquisition, Investigation, Methodology, Project administration, Resources, Supervision, and Writing.

Gerhard Navratil is a senior scientist in the Geoinformation Group at TU Wien. His research interest focuses mainly on data quality. His field of application ranges from land administration to navigation and augmented reality. His CRediT roles for this paper were Methodology

(transformations), Investigation (evaluation), Validation (quality assessment), and Writing (review & editing).

Markus Kattenbeck has been leading the Spatial-HCI lab in the research division Geoinformation at TU Wien since May 2019. His highly interdisciplinary research focuses on the behavioral correlates of human wayfinding, spatial cognition, and human computer interaction. His CRediT roles for this paper were Software, and Writing.

Negar Alinaghi is a PhD candidate in the Geoinformation Group at TU Wien. Her research focuses on Human Activity Recognition and GeoAI. Her CRediT roles for this paper were Software, and Writing (review & editing).

ORCID

Marcelo L. Galvão  <http://orcid.org/0000-0001-6204-3530>
 Paolo Fogliaroni  <http://orcid.org/0000-0002-0578-8904>
 Ioannis Giannopoulos  <http://orcid.org/0000-0002-2556-5230>
 Gerhard Navratil  <http://orcid.org/0000-0002-2978-5724>
 Markus Kattenbeck  <http://orcid.org/0000-0001-6028-0428>
 Negar Alinaghi  <http://orcid.org/0000-0001-6594-4481>

Data availability statement

The data, codes, and instructions that support the findings of this study are available at <https://geoinfo.geo.tuwien.ac.at/resources/>.

References

- Arth, C., *et al.*, 2015. Instant outdoor localization and slam initialization from 2.5d maps. *IEEE Transactions on Visualization and Computer Graphics*, 21 (11), 1309–1318.
- Billinghurst, M., Clark, A., and Lee, G., 2015. A survey of augmented reality. *Foundations and Trends in Human-Computer Interaction*, 8 (2–3), 73–272.
- Blut, C., and Blankenbach, J., 2021. Three-dimensional CityGML building models in mobile augmented reality: a smartphone-based pose tracking system. *International Journal of Digital Earth*, 14 (1), 32–51.
- Burkard, S., and Fuchs-Kittowski, F., 2020. User-aided global registration method using geospatial 3D data for large-scale mobile outdoor augmented reality. In: *2020 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*. Recife, Brazil: IEEE, 104–109.
- Cron, J., *et al.*, 2019. Head-mounted augmented reality visualisation. In: G. Gartner and H. Huang, eds. *LBS 2019: Adjunct proceedings of the 15th international conference on location-based services*. Vienna, Austria: ICA.
- Ghilani, C.D., 2010. *Adjustment computations: spatial data analysis*. 5th ed. Hoboken, NJ: John Wiley & Sons Inc.
- Hannah, B.M., 2001. *Modelling and simulation of gps multipath propagation*. Thesis (PhD). Queensland University of Technology.
- Hansen, L.H., *et al.*, 2021. Augmented reality for subsurface utility engineering, revisited. *IEEE Transactions on Visualization and Computer Graphics*, 27 (11), 4119–4128.
- Hedley, N.R., *et al.*, 2002. Explorations in the use of augmented reality for geographic visualization. *Presence*, 11 (2), 119–133.
- Hilliges, O., *et al.*, 2012. HoloDesk: direct 3D interactions with a situated see-through display. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '12*. New York, NY: Association for Computing Machinery, 2421–2430.

- Huang, W., Sun, M., and Li, S., 2016. A 3D GIS-based interactive registration mechanism for outdoor augmented reality system. *Expert Systems with Applications*, 55, 48–58.
- Huo, K., et al., 2018. Scenariot: spatially mapping smart things within augmented reality scenes. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18. New York, NY: Association for Computing Machinery, 1–13.
- Imottesjo, H., and Kain, J.H., 2018. The urban CoBuilder - a mobile augmented reality tool for crowd-sourced simulation of emergent urban development patterns: requirements, prototyping and assessment. *Computers, Environment and Urban Systems*, 71, 120–130.
- Lee, G.A., et al., 2012. CityViewAR: a mobile outdoor AR application for city visualization. In: *2012 IEEE international symposium on mixed and augmented reality-arts, media, and humanities (ISMAR-AMH)*. Atlanta, GA: IEEE, 57–64.
- Lee, J., et al., 2022. User preference for navigation instructions in mixed reality. In: *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. Christchurch, New Zealand: IEEE, 802–811.
- Lilligreen, G., and Wiebel, A., 2023. Near and far interaction for outdoor augmented reality tree visualization and recommendations on designing augmented reality for use in nature. *SN Computer Science*, 4 (3), 248.
- Low, C., and Lee, Y., 2014. SunMap+: an intelligent location-based virtual indoor navigation system using augmented reality. In: *International Conference on Frontiers of Communications, Networks and Applications (ICFCNA 2014 - Malaysia)*. Kuala Lumpur, Malaysia: IET, 1–6.
- Mazurkiewicz, B., Galvão, MdL., and Giannopoulos, I., 2023. BeeAR: augmented reality beeline navigation for spatial knowledge acquisition. *Proceedings of the ACM on Human-Computer Interaction*, 7 (MHCI), 1–17.
- Muhammad Nizam, S.S., et al., 2018. A scoping review on tangible and spatial awareness interaction technique in mobile augmented reality-authoring tool in kitchen. *Advances in Multimedia*, 2018, 1–14.
- Postert, P., Berger, M., and Bill, R., 2021. Utilizing CityGML for AR-labeling and occlusion in urban spaces. In: A. Kamilaris, V. Wohlgenuth, K. Karatzas and I.N. Athanasiadis, eds. *Advances and new trends in environmental informatics*. Cham: Springer International Publishing, 209–223.
- Qiao, X., et al., 2019. Web AR: a promising future for mobile augmented reality - state of the art, challenges, and insights. *Proceedings of the IEEE*, 107 (4), 651–666.
- Qiu, X., et al., 2023. Impact of AR navigation display methods on wayfinding performance and spatial knowledge acquisition. *International Journal of Human-Computer Interaction*, 0 (0), 1–21.
- Rao, J., et al., 2017. A mobile outdoor augmented reality method combining deep learning object detection and spatial relationships for geovisualization. *Sensors*, 17 (9), 1951.
- Rauschnabel, P.A., et al., 2022. What is XR? Towards a framework for augmented and virtual reality. *Computers in Human Behavior*, 133, 107289.
- Rovira, A., et al., 2020. Guidance and surroundings awareness in outdoor handheld augmented reality. *Plos One*, 15 (3), e0230518.
- Sahu, C.K., Young, C., and Rai, R., 2021. Artificial intelligence in augmented reality - assisted manufacturing applications: a review. *International Journal of Production Research*, 59 (16), 4903–4959.
- Singh, A.K., et al., 2021. Virtual global landmark: an augmented reality technique to improve spatial navigation learning. In: *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI EA '21. New York, NY: Association for Computing Machinery.
- Stachniss, C., Leonard, J.J., and Thrun, S., 2016. *Simultaneous localization and mapping*. Cham: Springer International Publishing, 1153–1176.
- Taketomi, T., Uchiyama, H., and Ikeda, S., 2017. Visual SLAM algorithms: a survey from 2010 to 2016. *IPSJ Transactions on Computer Vision and Applications*, 9 (1), 1–11.
- Wang, L., et al., 2016. Smart device-supported BDS/GNSS real-time kinematic positioning for sub-meter-level accuracy in urban location-based services. *Sensors*, 16 (12), 2201.
- Wang, Y.J., et al., 2017. Augmented reality with image registration, vision correction and sunlight readability via liquid crystal devices. *Scientific Reports*, 7 (1), 433.

- Wu, Y., Che, W., and Huang, B., 2021. An improved 3D registration method of mobile augmented reality for urban built environment. *International Journal of Computer Games Technology*, 2021, 1–8.
- Zollmann, S., et al., 2021. Visualization techniques in augmented reality: a taxonomy, methods and patterns. *IEEE Transactions on Visualization and Computer Graphics*, 27 (9), 3808–3825.