IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 6 June 2024, accepted 25 June 2024, date of publication 28 June 2024, date of current version 17 July 2024.

*Digital Object Identifier 10.1109/ACCESS.2024.3420732*

RESEARCH ARTICLE

# Reliability and Accuracy of Indoor Warehouse Navigation Using Augmented Reality

**PETR HOŘEJŠÍ, TOMÁŠ MACHÁČ, AND MICHAL ŠIMON**
Department of Industrial Engineering and Management, University of West Bohemia, 301 00 Pilsen, Czechia

Corresponding author: Petr Hořejší (tucnak@fst.zcu.cz)

This work was supported by the Internal Science Foundation of the University of West Bohemia through the Environmentally sustainable production under Grant SGS-2023-025.

**ABSTRACT** Augmented reality (AR) plays a crucial role in Industry 4.0, serving as a foundational element in various stages of the product and process lifecycle. AR is employed in both outdoor and indoor environments, utilising technologies such as GPS, SLAM, RFID, Wi-Fi, and inertial sensors for positioning and navigation. This study focuses on assessing the reliability, functionality, and accuracy of indoor warehouse navigation using inertial sensors commonly found in smartphones. Standard AR development tools like ARCore, AR Foundation, and ZXing were chosen for validation. A mobile application was developed using Unity3D. The field research was conducted in a medium-sized warehouse. Results showed an 83% success rate in reaching the target with an average accuracy of 0.48 metres. Development iterations addressed issues identified during testing including navigation failures and instability caused by warehouse lighting conditions. Comparisons with other technologies revealed AR's competitive advantages, despite some existing challenges. The study concludes that ongoing innovation in hardware and software will likely overcome current limitations, paving the way for broader AR adoption in industrial settings.

**INDEX TERMS** Accuracy, augmented reality, industry 4.0, indoor warehouse, navigation, reliability.

## I. INTRODUCTION

Industry 4.0 has brought a fundamental shift in the paradigm of industrial production [1]. This evolution is characterised by improvements in industrial processes in the areas of manufacturing, engineering, materials management, supply chain and product life cycle management [2]. In Industry 4.0 we encounter technologies such as the Internet of Things, artificial intelligence, robotics, big data, and cloud storage. Other advanced technologies are also used such as virtual and augmented reality [1]. Deloitte [3] characterises virtual and augmented reality as key elements of the Industry 4.0 concept, and they are technologies which are growing exponentially in terms of process and product. Weyer et al. [4] highlight aspects of Industry 4.0 such as smart product, smart machine, and the augmented operator. These agents use technical support as the main interaction technology in the form of a tablet or smart glasses with augmented or virtual reality.

Augmented reality is a technology that conveys a polysynthetic world to the user [5]. Augmented environments are the interconnection of the real world with virtual objects that complement reality [6], [7], [8]. Azuma [5] lists the possibilities that may be found in future augmented reality deployments, such as maps for navigation and visualisations of current and future environments. Augmented reality systems are limited to a portion of the user's field of view (FoV) [9]. AR technology uses several techniques including marker-based AR and markerless AR [10]. These augmented reality systems can be applied in areas across industries such as service, manufacturing, sales and marketing, design, human resources, and training [11]. Augmented reality is also used in many other areas, for example healthcare, military applications, shipbuilding, and aviation [11], [12]. Santi et al. [13] and [14] include among the main areas of Industry 4.0 the deployment of augmented reality in logistics support, examples being Pick by Vision,

The associate editor coordinating the review of this manuscript and approving it for publication was Lorenzo Ciani.

operator guidance in item selection and warehouse worker training and induction. The area of logistics and augmented reality has been addressed by Broum et al. [15], who connected an AGV system with AR using Vuforia SDK and FieldBit from Help Lightning. The output was an application that visualised information about the energy consumption of each AGV. Another area for application in industry is manufacturing, specifically as support through expanded assembly instructions [16]. With the rise of AR, navigation applications have emerged [17]. For outdoor AR navigation, GPS technology is used to determine the location [18]. If we are to create AR navigation for indoor spaces, GPS is not ideal as the geolocation of the satellite signal does not penetrate well through the structure of the building [19]. ORB-SLAM technology is found in outdoor navigation, but also in indoor navigation [20], [21]. Typical methods used for indoor positioning are Bluetooth and Wi-Fi [22]. Other methods can include ultrasonic and UWB signals, or inertial sensors [23], [24]. Traditional navigation systems mainly allow procedural and navigation information [11]. In navigation, POI (Point of Interest) has an important role. These points of interest represent key locations in the proposed structure [25]. Augmented reality systems integrate navigation instructions directly into the environment, allowing for smoother navigation and increased user alertness. This suggests that AR improves route understanding and facilitates route memorisation [26]. Standard navigation systems display abstract navigation information such as arrows or a bird's eye view, but they lack sufficient immersion of the user in the AR environment and appear less intuitive [27]. We can also find mobile applications that combine semantic technologies with augmented reality, where physical objects are combined with digital content [28].

### *A. RELATED WORK*

Zhang et al. [22], [29], and [30] use simultaneous localisation and mapping (SLAM) in their system, which detects visual features in the form of points in the acquired camera images. These points can detect the change in the user's location. Based on their experiments, it was confirmed that SLAM based applications are much more accurate than GPS based applications. Ayyanchira et al. [31] developed an indoor navigation system that works with devices relying on a camera capturing the environment. The hardware devices chosen for the system were an iPhone, an iPad, and HoloLens glasses. The system is complemented with a WIM (world-in-miniature) that works as a support for navigation. Indoor tracking systems are most often based on communication technologies or image detection algorithms. Pirzada et al. [32] describe that active tracking is optimal for solutions where accurate tracking and registration is required. In contrast, passive tracking is computationally intensive. This categorisation stems from the technology employed, which deals with location localisation possibilities. Specific examples of active tracking include radio frequency tracking systems, which are typically quite expensive and can vary in accuracy due to interference, equipment quality, or other environmental factors [33]. Passive tracking, such as radio tomographic imaging (RTI) or Device-Free Passive (DFP) localisation using an array sensor, uses an antenna array on the receiver to enhance localisation accuracy [33], [34]. Several active tracking solutions, such as Bluetooth beacons, can also be integrated with passive localisation techniques, resulting in a flexible hybrid solution [36]. Shewail et al. [37] describe indoor tracking systems as a diverse field that can be divided into two main approaches: image-based detection algorithms and communication technologies. The communication technologies used in indoor navigation include GPS, Wi-Fi, Bluetooth, RFID, UWB, Infrared, ZigBee, Cellular, VLC, NFC, FM and Ultrasound. The detection algorithms are FAST, SURF, SIFT, FAST-SURF and ORB-SLAM. If we are not focusing on the basic approach to localisation but rather on the use of technologies within a comprehensive system design, an example of combining the detection algorithm visual SLAM with Bluetooth is suggested by Kao and Huy [38]. A more precise description of the UWB-based navigation concept is described by Shi and Ming [39]. Augmented reality provides support to users when navigating indoors, especially when the route is longer. It improves the efficiency of navigation and brings positive acceptance regarding visual access, user comfort, and a sense of support if encountering a dangerous obstacle or building evacuation [40]. For longer indoor routes (greater than 200 metres), common AR algorithms cannot be used. It is advisable to use a combination of technologies with additional location information, either through user interaction or an external system [41]. Conventional route selection algorithms are not suitable for routes longer than 200 metres because the communication capacity gain surpasses the increase in route length [42]. Moreover, conventional routing algorithms are insufficient for large road networks due to their inability to compute optimal routes efficiently [43]. Elgendy et al. proposed an AR system based on three components: marker detection, location calculation and navigation. The disadvantage of marker-based systems is the risk of occlusions: not retrieving markers, or partial occlusion [44]. A Wi-Fi and NFC independent application has been proposed by Verma et al. [45], similarly by Ayyanchira et al. [31], for the iOS operating system, which uses ARKit for development in a developer environment. The frameworks and tools included are XCode, Firebase and Three.js [45]. Liu and Meng [19] focused on indoor navigation to improve the user's spatial learning with the design of an interface for HoloLens glasses. From the research, it can be concluded that it is more beneficial for the user to add icons as well as textual information to serve as virtual landmarks in the application interface. The issue of the user-friendliness of the user interface for the HoloLens heads-up display has also been addressed by Keil et al. [46]. Ng and Lim [47] created an indoor navigation system using the IndoorAtlas SDK and MapCreator2. They also determined the user's current position in the ARCore SDK link to display augmented reality navigation and the corresponding database. The mobile phone primarily works with a magnetic field, Wi-Fi signals and inertial sensors to determine its position and pathfinding via the IndoorAtlas SDK [47]. Corotan and Irgen-Gioro [48] extended the navigation system to include a real robot that guides the user to a target position. The robot is connected to a mobile device through which the navigation takes place. The technology used is Bluetooth and an Arduino Uno microcontroller. Michel et al. [49] confirmed the effect on the accuracy of different sensors that are commonly found in every mobile phone, an accelerometer, a gyroscope and a magnetometer. The authors recommend the use of custom calibration and algorithms instead of those provided by the mobile phone operating system, thus increasing the portability and stability of navigation. Ehrlich and Blankenbach [50] address inaccuracies in pedestrian navigation, where the movement of a smartphone user, determined by built-in sensors, undergoes filtration via a particle filter in conjunction with a building model, enhancing localisation accuracy through optimisation using WLAN fingerprints and Bluetooth signals, notably integrating BLE for higher precision. Huang et al. [51] created an application for indoor use with beacon technology. They mention inaccuracy of up to 5 metres as a drawback of the system. Dosaya et al. [52] applied audio recognition functionality to their indoor AR navigation system. As the user gets closer to the target the sound frequency increases and decreases as the target recedes. Yang et al. [53] developed a semantic-driven method for indoor navigation from RGB-D sensor data reconstruction. Similarly, Chidsin et al. [54] proposed a system that uses an RDB-D camera to observe the environment and relies on ORB-SLAM for mapping and positioning.

One of the main drawbacks of the proposed AR systems based on various technologies is the inherent inaccuracy of these systems. This results in differing outputs between reality and virtuality, meaning that environments are not localisation-identical [55], [56]. The inaccuracy of AR solutions is typically in the range of a few metres [22]. For instance, the application developed by Huang et al. [51] exhibits inaccuracies between 3 and 5 metres. Orfanos et al. [57] focus on assessing the Wi-Fi RTT range quality, with 2D-space ranging tests achieving an average root mean square error (RMSE) of 1.1 metres across various devices. Guo et al. [58] integrated data from Wi-Fi and MEMS-IMU for smartphone positioning, with experimental results from tests demonstrating an average positioning accuracy of 0.572 metres.

### B. DEFINING THE RESEARCH QUESTION

Based on our literature research, we find that there are many different navigation systems that apply augmented reality technology. In these studies, the researchers mostly encounter low reliability and accuracy, thus limiting the functionality of the system. The most widely used technology for implementing AR navigation is SLAM and inertial sensors. These variants are also the simplest in terms of usability on a mobile phone or tablet, and no additional hardware is required.

In this study, we want to demonstrate whether the proposed system achieves reliability and speed when using it under real conditions in industrial practice. In other words, we investigate the reliability and related practicality of the algorithms that are included in the standardised libraries. The research question can be formulated as follows:

Q1: Is it currently possible to achieve target guidance accuracy with a 99% success rate using augmented navigation based on inertial sensors and standardised AR Foundation and ARCore frameworks?

## II. METHODOLOGY

The defined aim is to validate and use augmented reality navigation in an indoor warehouse. The software packages selected for the verification are described. Standardised packages are selected, so this study can be seen as a verification of the usability of a combination of these standardised AR development packages for indoor navigation.

With increasing urbanisation, indoor spaces are becoming larger and more complex **[53]**. In view of this, a medium-sized representative space is used for the research. The potential of AR navigation lies in the current trend in the demand for storing goods and then picking them in small shipments, a trend that can be found, for example, in e-commerce sales. The human factor still plays a key role in the picking method, so we want to focus on this element, which is prone to error, disorientation in space and not finding the optimal path. It is precisely these problems that augmented reality-enabled navigation can help with.

In the following subchapters, first the software tool for testing is introduced and then the methodology itself:

1. The development environment is the initial part of the structure and forms the basic development phase. For this, the widely used Unity3D development environment is used. We also use standardised libraries created for augmented reality needs. In the Unity3D environment we create all the virtual objects that are further applied in our case to the mobile device and are adaptable to other devices that support augmented reality features.
2. The initial development iteration is the first version of the designed application in the development environment. It is an unoptimised pilot version of the application. Based on the identified problems, the version can be corrected to an improved (‘‘sharp’’) version. We repeat this process to produce an optimised version. This is the development cycle of the application on which we test the augmented reality methods. The proposed key algorithms are also rebuilt.
3. Lastly, the actual methodology for testing the reliability of navigation within the software packages is presented.

### A. DEVELOPMENT ENVIRONMENT

At this point, it is useful to first state what software (framework) is used for the application and thus the study. As already mentioned, the aim is to use standardised software packages, especially to allow further comparison of similar applications and to easily build on the research. It is also important to list the different versions of the software, as any future higher versions may of course affect the outputs. The following packages were applied to implement the reference software used later in the study:

**Unity3D (2022.3.3f1)** – originally a multiplatform game engine created by Unity Technologies. This engine is now also used for a wide range of non-gaming applications. It is possible to develop 2D and 3D applications for a variety of platforms - PC, console, Android, iOS, or web interface. For scripting, Unity supports the C# scripting language [59].

**AR Foundation (5.0.6)** – a cross-platform framework for creating augmented reality applications. AR Foundation does not directly create augmented reality elements, but only mediates them. Subsystems are required to implement AR elements [60].

**ARCore XR Plugin (5.0.6)** - a software development kit from Google that makes it possible to work with physical augmented reality artefacts. Using APIs, this toolkit has motion tracking, environment recognition and light estimation [61].

**ARCore Depth API** – the interface is part of the ARCore SDK and helps the camera understand the size and shape of real objects in the scene, bringing a more realistic representation of objects into the virtual scene. The ARCore SDK integrates a machine learning element that can recalculate the current position of the device and the displayed AR content in combination with other sensors - gyroscope, magnetometer and accelerometer - in case of occlusion or image failure [62].

**ZXing. NET (3.5.1)** – library supporting decoding and generation of barcodes - QR Code, PDF 417, EAN, UPC, Aztec, Data Matrix and Codabar in the form of images.

The connection between Unity3D, AR Foundation and ARCore is standard when creating applications for Android operating systems. If we are creating apps for the iOS operating system, we need to replace ARCore with ARKit, which is only for Apple devices [63].

### B. INPUT DEVELOPMENT ITERATION

The application can determine the location in the warehouse, navigate the user to the item and implements a marker reading system for calibrating the system and confirming that the item has been found. The application includes a minimap for the user and a series of buttons that trigger individual functions such as creating a simulation issue slip, manual entry of an item and the ability to scan a QR code.

#### 1) APPLICATION REQUIREMENTS

We defined the requirements that the application must meet. These are navigation using augmented reality displays, applicability in the internal warehouse environment, support for commonly available devices such as phone or tablet, simulated picking - to several warehouse positions, visual accessories - marking the target position, simple and intuitive handling, direction to the target defined by a virtual line and the possibility of selecting several navigation modes. The application is designed so that most of the work is done by the libraries mentioned above, yet it was necessary to add a significant amount of service code.

A real indoor warehouse at the company site was chosen for testing the application, but the limits of this application are determined by the predefined environment for which the application is designed. The model of the specific environment is a direct part of the application. The application must be modified if we want to use the application in another environment.

The test environment is a warehouse for storing clothes and shoes. The shelves are divided at the bottom of the racks, where the manual picking is done. In the upper part of the racks, the unloading is done by a forklift. The height of the racks is 10 metres, and the length is 25 metres. The width of the rack positions is 2.8 metres and 3.6 metres. For lighting, skylights are placed in the warehouse in an irregular layout with pendant lights and windows on the west side.

#### 2) APPLICATION DEVELOPMENT

At the beginning it is necessary to create a virtual reference model in Unity3D for tracking in AR. A calibrated Stabila LD-320 laser rangefinder was used to create the 3D model. A 2D drawing based on the measurements was created in Autodesk Inventor Pro.

Based on the drawing, a NavMesh was generated in Unity3D, which is generally used to define the area for object movement. In our case, this mechanism is mainly used to generate the navigation line. After initial tests, it was found that the navigation line was not clear at the locations of the shelf edges. These edges were "wrapped" with a region with a defined lower preference for navigation, this way the navigation line does not break "sharply" at the edges of the shelves (see **FIGURE 1**) and looks more natural. 3D artifacts must then be generated at the target position to inform about the successful finding of the target. Tags (here QR codes) were used for position retrieval and calibration.

The user works only based on the technologies available in conventional mobile devices - magnetometer, gyroscope, and accelerometer. For this reason, it is necessary to know the initial reference position before starting any navigation (this may not be true for repeated navigation). Repeated navigation is defined as an already configured system that is calibrated with the real environment and has already undergone the initial navigation to a given position (destination). After completing this initial navigation, it is possible to proceed to further navigation without the need for an initialisation step, i.e., without the requirement of performing input calibration. Calibration options have therefore been implemented in the software; either calibration to a predefined default position or calibration by scanning a marker associated with

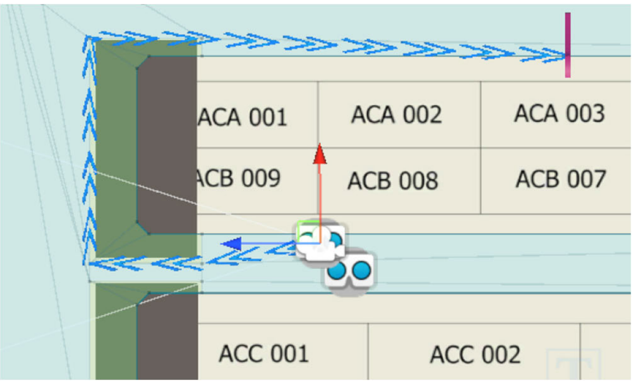


**FIGURE 1.** Correction of the generated route in the rack corner area for NavMesh.

a specified rack. The free on-line QR Planet generator was used to create and manage the QR codes.

### 3) DEVICE

The app was tested on a Samsung Galaxy A52 5G device that is supported by the ARCore Depth API. The device parameters used for testing can be said to be typical for a mid-range mobile. For new smartphones, ARCore Depth support is absolutely essential [64]. The operating system of the test mobile phone is Android 11, and the display has a refresh rate of 120 Hz. The resolution of the main camera is 64 Mpx and the battery capacity is 4500 mAh. The method works primarily on inertial sensors, an accelerometer, a gyroscope sensor, and a magnetometer. These sensors are included in all newer mobile phones [65].

### 4) APPLICATION PRINCIPLE AND USER INTERFACE

We have created essential elements for the GUI, such as a minimap for orientation in space [66]. The aim was to create a pleasant, simple, and clear user interface that would not obstruct the view of the camera image. The default view when the application is launched is the initial calibration screen. Once calibrated the user automatically switches to the view for manual navigation mode. The picking simulation is also an option in this view. The app is designed primarily for a 1080 × 1920 mobile phone in portrait mode. The Unity3D software works with a 1:1 warehouse layout where the default position is automatically set when the application starts. This means that we have to stand in the right place in the warehouse and then confirm the system calibration, otherwise the system will desynchronise. Alternatively, the second option is to scan the nearest position on the shelf via ZXing.NET (library for working with QR codes) and thus also perform the input calibration. This step must be done as accurately as possible, as the accuracy of the entire system depends on this point. When testing, we must consider that in practice the worker does not always perform the calibration correctly. At the input, either an input calibration is performed using a predefined position or using a QR code. Simply pointing the mobile phone at the QR code will perform a calibration of the system including the correct angle and direction.

The first time the application is started, the system calibration initialisation step is performed, as shown in the flowchart in **FIGURE 2.** The flowchart starts with the "Application launch" event and then divides into two paths:

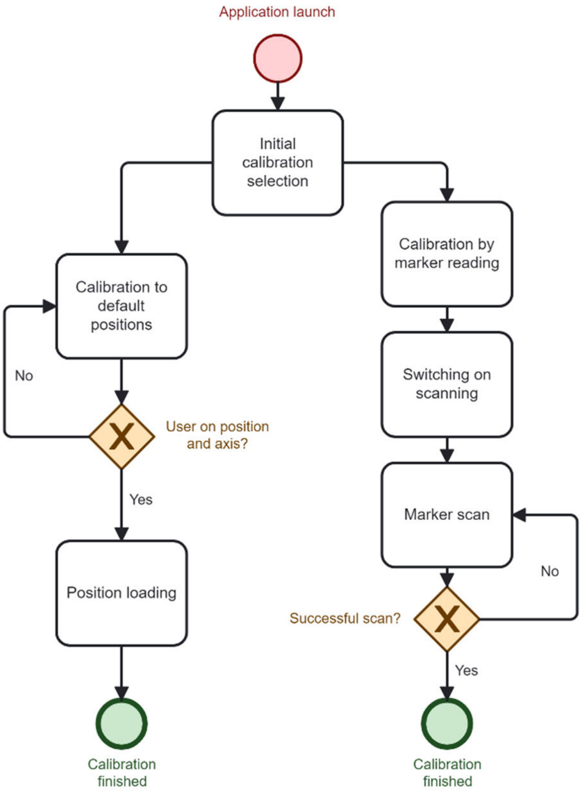


**FIGURE 2.** Calibration of position.

### 5) INITIAL CALIBRATION SELECTION

This path continues to a decision point that verifies that the user is in the desired position and axis. In the case of a negative answer, the diagram returns to the beginning. In the case of a positive response, it continues to "Position Loading" and ends as "Calibration finished".

### 6) CALIBRATION BY READING THE MARKERS

This path goes through steps such as "Marker Scan", followed by another decision point. A successful scan leads to an indication that calibration is complete, while an unsuccessful scan leads to a return to "Marker Scan".

The algorithm is configured to allow two approaches. The first one involves manually selecting the target destination. The specific process is illustrated in **FIGURE 3**, describing the Manual Target Selection workflow. It begins with the "Manual Target Selection" phase, followed by position calibration to ensure accuracy. The system then checks if the calibration is successful. If it is, it proceeds to scan the marker; otherwise, it returns to the "Position Calibration" phase. After scanning the marker, a target is selected based

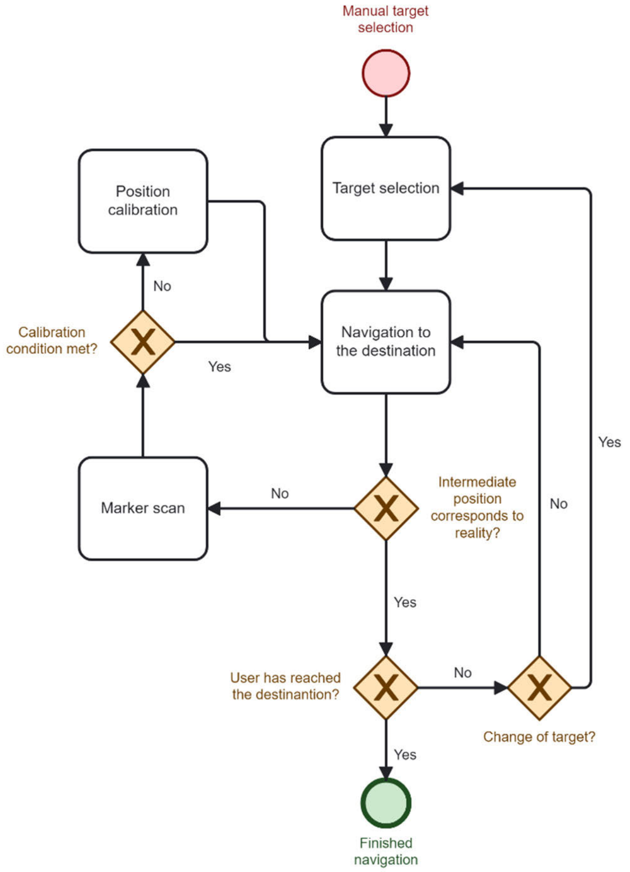


**FIGURE 3. Manual target selection.**

on the acquired data for navigation. Subsequently, navigation to the target begins, during which position checks and corrections are performed regularly if necessary. Upon reaching the target, the system indicates success to the user. In case of failure to reach the target, a change of target or further corrections can be made. The entire process is governed by a loop that repeats until the user reaches the final destination or a change in destination occurs.

The second possible approach involves a pick simulation. The algorithm for the simulation is depicted in **FIGURE 4**. It begins with "Picking Simulation", leading to "Assignment of Target" and then continues to "Navigation to the Destination", followed by "Marker Scan." After "Marker Scan" the decision points are: if a condition is not met, proceed to "Calibration" and if calibration is successful, return to "Marker Scan". If calibration is not needed or is successful, it moves on to the next decision point, which queries whether the marker corresponds to the desired position.

The workflow of using the user environment while navigating to the target can be seen in **FIGURE 6**.

Figure 6A shows the standard visualisation during the user's progress to the target in manual mode. The first mode is manual item selection. The user selects the position they want to navigate to in the menu. The menu of positions is in a drop-down list. The list is created for specific positions.

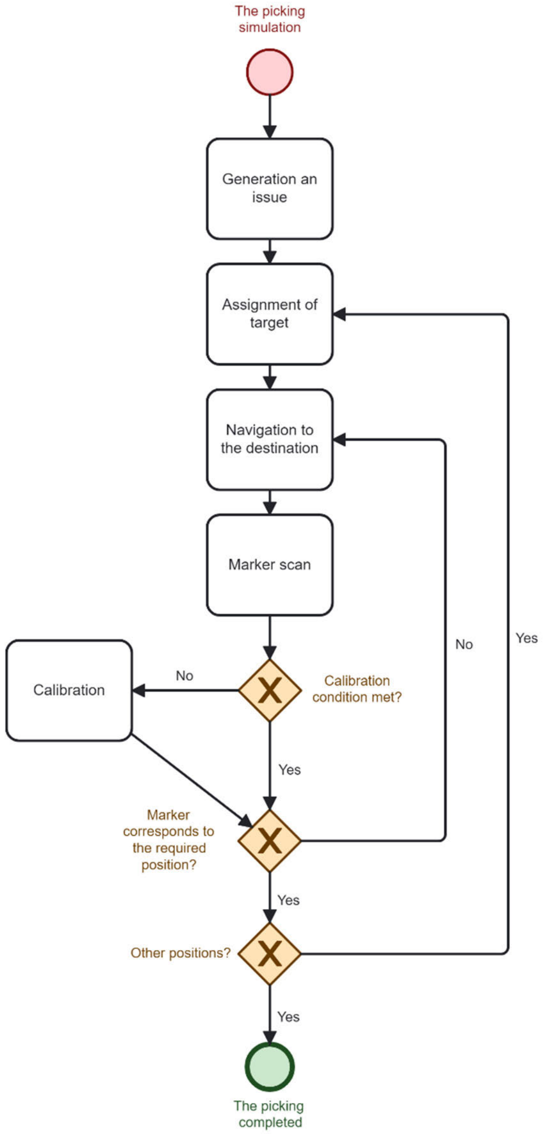


**FIGURE 4. Simulation of picking.**

After selecting the position, it is confirmed (in Czech: 'Zobrazit navigaci'). The display shows a navigation line that guides the user to the destination. In this step, the displayed line is followed. When we are close enough, we can switch the display to "scan" and retrieve the position of the item. If the position does not match, we can recalibrate the system and proceed further. If the user does not reach the target, they have the option to change the target and the process can be started again. If the user navigates to only one selected target, the target is named in the yellow window with the title (in Czech: 'Další cíl je'). Figure 6B shows the approach to the target. At this point, a virtual purple arrow is shown to the user, indicating the exact position. When the user is close enough to the position, the user can switch again to the "scan" option. In the scan mode, a target is picked up (Figure 6C), the user points the mobile device at it and reads the QR code. The target is then confirmed to have been found.

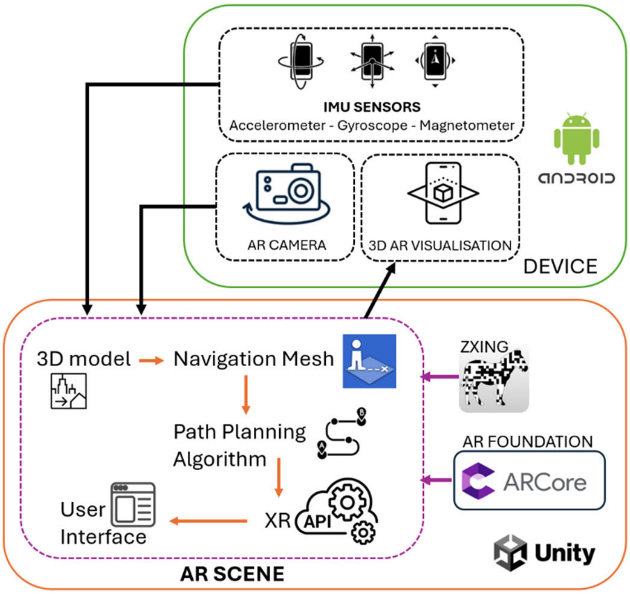


**FIGURE 5.** Summary design of software framework.

**FIGURE 6.** User interface: A - manual mode, B - visual target identification (CÍL means TARGET), C - marker scanning.

For performing further tests, two modes were created in the application. Manual mode, where the target location can be selected from a list, is described above, and the second mode which generates a random list of target locations. The picking simulation mode works according to the algorithm. The picking simulation is created for ten randomly generated stock positions. This mode can be selected using a button (in Czech: 'Zobrazit výdejku'). This automatically starts the generated list of items we need to track down. Navigation is again along the displayed line which leads us to the target. The QR code is scanned at the target. After a successful scan, the next position appears. If the position does not match (the minimap helps us to do this) we recalibrate. The process is repeated for the remaining targets until we have the whole order.

In conclusion, the **FIGURE 5** presents a framework for an augmented reality system on Android devices. This developed framework describes the basic structure of the proposed system and provides an outline of the design concept.

### C. TESTING METHODOLOGY

After several initial tests and several development iterations, a debugged software tool was available to test the reliability of the combination of the AR packages.

For testing, an order picking form is created in the application, which always contains 10 different items that are randomly generated for testing purposes. In each order, different positions are generated each time for navigation. Testing was performed both during the development of the application and after the prototype was built. Navigation and orientation functionality in space could only be tested in the warehouse. The scenario of randomly generating order items was compared for all the iterations equally.

The purpose of the testing is to demonstrate the accuracy, reliability and stability of the proposed solution that combines standard approaches. The overall behaviour of the application in guiding the user to the destination was also investigated. A total of 30 test runs were performed in the first development iteration, and 71 runs were performed in the sharp version, with an average distance travelled of 29.38 metres.

The deviation measurement was performed by comparing the position of the displayed virtual target arrow (its position is determined by the coordinate of the warehouse position) and the physical location of the warehouse position (marker). The distance was measured with a manual tape measure, which affects the accuracy of determining the deviation. A textual commentary was also prepared for each route pass. Any non-standard situations that occurred during the route traversal were recorded. Such situations can be basically divided into two groups: critical errors that lead directly to the user getting lost and not reaching the destination, and non-critical errors that do not cause serious failure or loss of position such as failure to display the navigation line and re-display.

The assessment of whether a user has ''got to the destination'' depends on the subjective assessment of the individual. The criterion for determining whether the user has reached the destination is the user's visual confirmation that the search product matches the location (which is the destination). In addition, the application is equipped with a QR code scanning function that confirms the completion of the navigation process. If the QR code does not match at the destination and the app displays an ''error message'', it means that the user has not reached the destination.

The experimental testing methodology can be succinctly summarised as follows: Prior to testing, positions are marked with QR codes. The application then calibrates based on these codes or a default position. This is followed by a simulation of order selection, during which the system generates random positions. Subsequently, navigation commences using a 3D AR navigation line directing towards the next positions. Upon servicing all positions, the system concludes navigation, ready for the next simulation cycle. This process evaluates the stability and accuracy of navigation across various developmental stages, providing data for further application refinements.

## III. RESULTS

The pilot testing focused on investigating the accuracy, the stability of the solution and the behaviour of the application in guiding the user to the target. The aim of the testing was to obtain data for improving the application and optimising it. Based on the outputs, a new version of the software was created and tested further. The application is only a tool for testing the different methods.

The following shortcomings were found when using the input development iteration:

- Starting only at the initial calibration position, navigation can only be started from one location.
- The navigation line is very high off the ground.
- The navigation line crosses the corners of the rack.
- The user is not informed which position they are being guided to.
- A virtual scene restart and subsequent failure of the camera autofocus function was triggered during scanning, and therefore the camera is constantly focusing.

Changes have been made to lower the navigation AR line of the Y coordinate, so the line is generated at a defined height based on the environment recognition, the NavTrajectoryAdjustment element has been assigned to the NavMesh command. A command was added to turn on the camera autofocus after scene restart. Information elements were added for a more user-friendly interface – a notification button on the pointing position and a position information board. A modification to force the AR camera autofocus on/off, which causes the camera to constantly focus. There is a possibility to calibrate to the storage position. Calibration can be performed from locations other than the starting position.

### A. 2nd DEVELOPMENT ITERATION

Reliability testing of the solution has been done for this second iteration. The result of the measurement was a 73% success rate for guiding to the target. Out of 30 attempts, 22 attempts were successful in reaching the target. At the same time, there was a large inaccuracy of navigation to the target with an average deviation of 0.45 metres. 50% of the routes showed a deviation greater than 0.3 metres. The standard deviation is 1.12 metres. The conclusion of the first test shows significant inaccuracy, as can be seen in **TABLE 1.**

**TABLE 1. Output of the second development iteration application test.**

| Number of attempts | Getting to the destination | Average deviation | Standard deviation | Continuous navigation failure |
|---|---|---|---|---|
| 30 | 22 (73%) | 0.45m | 1.12m | 20% |

During testing, many instances of problems were noted at the east wall of the warehouse, where the space is relatively narrow at 1.3 metres wide and shaded. This area of the warehouse has inadequate lighting. In addition, there are stacked boxes against the wall, creating a more complex and even narrower environment. The failures were mostly due to a slight deviation of the device position in the virtual scene which, due to the narrow space (small distance of the user from the shelf), caused the virtual position to be outside the NavMesh area and thus a navigation route could not be generated. If the user continues moving in the correct direction, the navigation is displayed again (the virtual position returned to the NavMesh desktop).

Changes were made to remove the automatic calibration after scanning the marker, as the calibration is error prone. Calibration is set to be automatically conditional on the number of scans without calibration or triggered by the user. Scanning the QR code for calibration is done repeatedly, but only if the QR code is still in the scene. The interface has been improved to add a user directional arrow to the minimap. This provides better orientation for the user instead of the original dot. The height of the generated navigation line is still not satisfactory.

### B. INNOVATED APPLICATION

The changes have brought improvements. We achieved fewer errors and more stable navigation by removing automatic calibration. The new version has slightly improved navigation capability. The navigation reliability increased from 73% to 83%. Out of 70 attempts, 59 reached the destination. The increase in navigation reliability is 10%. The standard deviation decreased to 0.91 metres. These results can be attributed to the elimination of excessive position calibration.

The current average deviation is less than 0.5 metres. This is summarised in **TABLE 2.**

**TABLE 2.** Output of the innovated application test.

| Number of attempts | Getting to the destination | Average deviation | Standard deviation | Continuous navigation failure |
|---|---|---|---|---|
| 70 | 59 (83%) | 0.48m | 0.91m | 26% |

The warehouse is illuminated by skylights and windows on the west side, from which bright light penetrates the hall. Passing through an aisle in which the light was only from windows in the wall and pendant lights, without a skylight, this combination created a tunnel effect. Even for the human eye, it was difficult to focus on details at times, as the glossy floor and painted surfaces of the shelves greatly reflected the sunlight. Conversely, if the route leads away from the windows towards the wall and the aisle has a skylight, the lighting conditions are much more favourable, and the details of the painted surfaces are more distinct.

A problem in testing repeatedly came up when moving against a bright backlight. In such cases, the application had trouble correctly determining the distance covered and often resulted in random bouncing of the guideline and subsequent incorrect determination of the rack position. In such cases, a large part of the display image is over-illuminated. Target points are then difficult to detect. In certain situations, the application was able to catch this error and calculate its position, but only when the user moved at least one metre away.

Following the second development iteration, it was determined that recalibration after scanning the QR code is unnecessary, as the algorithm's stability sufficed for subsequent navigation. The original version of the application included automatic calibration, which was removed in the second development iteration. This adjustment stemmed from findings in previous testing, indicating error-prone repeated calibration (with each QR code scan) due to high non-convergence between real and virtual environments. Automatic calibration was replaced with conditional calibration based on the number of scans without calibration or user-induced calibration, thereby reducing the risk of human-induced position and rotation miscalibration. It was confirmed that calibration based on a conditional scan count leads to a more stable and accurate algorithm, enabling traversal of three to five storage positions without additional calibration.

Many applications using ARCore and ARFoundation libraries include customisations such as implementing proprietary object recognition algorithms, performance optimisations, user interface adjustments, and integration with other technologies. This approach enables the achievement of specific functionalities that standard libraries may not offer. If we focus on a general level of implementation for this software framework in various environments, it is not feasible. The algorithm is specifically designed for a particular virtual environment, meaning that any minor change in the real environment can easily disrupt compatibility with virtual objects.

**FIGURE 7** is a graph showing the reliability scores for each route subjected to experimental verification. Reliability was categorised into three levels: Low, Medium, and High. The Low level indicates a route with a critical error, where the user did not reach the target destination. The Medium level signifies that the user reached the target destination, but a non-critical error occurred during navigation. The High level is the best, indicating no errors in navigation and the user reached the destination without issues. The overall reliability score, ranging from 0 to 3, was then calculated. For the second development iteration, we are slightly above the median level.

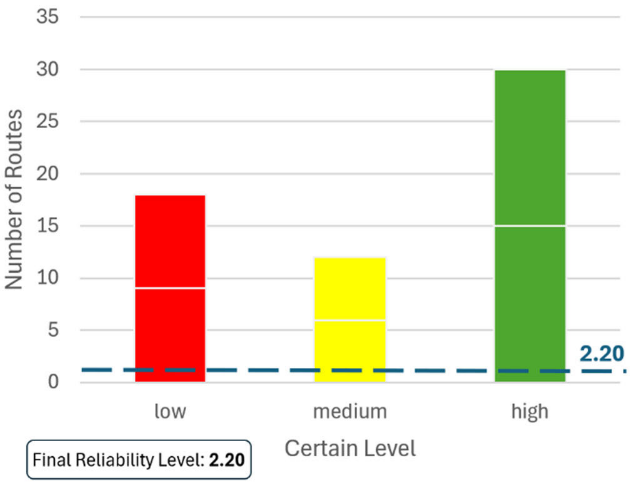


**FIGURE 7.** Levels of certainty for $2^{nd}$ development iteration.

We applied the same approach used in the second development iteration to evaluate the reliability level of the innovative application, the results of which are depicted in **FIGURE 8.** The resulting reliability level is higher than in the second development iteration, with the value increasing from 2.20 to 2.41. This level represents reliability almost halfway into the High category.

The overall view of all the resulting reliability levels can be seen in **FIGURE 11.** Here, we can clearly observe how the overall level has improved over the development iterations. The outcome suggests that there is still room for improving and enhancing stability, which will contribute to the overall reliability rate.

**FIGURE 10** and **FIGURE 9** show the Cumulative Flow Diagrams (CFD) for each development iteration. Each diagram features four lines: the blue line represents the cumulative count of goal achievements, the green line indicates the number of faultless navigations, the yellow line shows non-critical navigation states, and the red line depicts critical states of failing to reach the target position. The diagram (see **FIGURE 10**) indicates good performance with some variability, demonstrating that tasks are generally completed successfully with a manageable number of errors. In contrast,

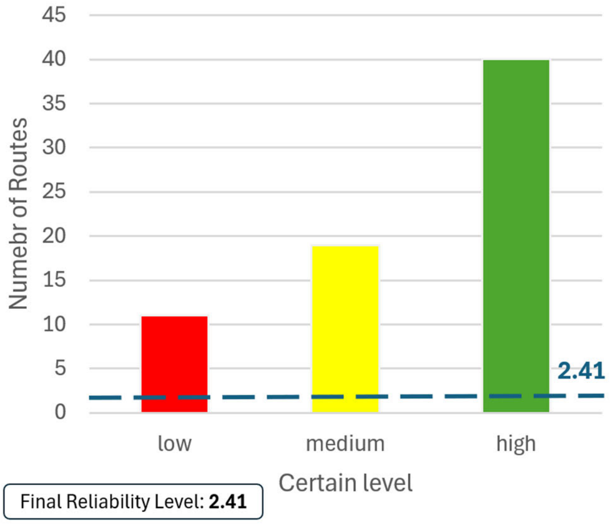


**FIGURE 8.** Levels of certainty for innovated application.

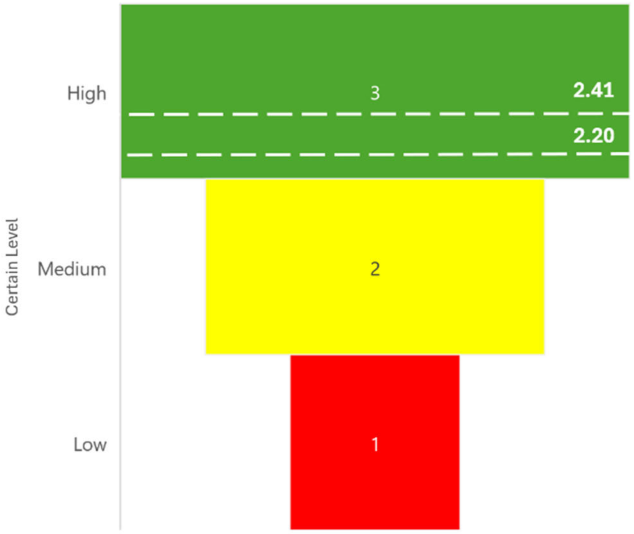


**FIGURE 9.** Overall level of certainty.

the diagram (see **FIGURE 9**) demonstrates a robust and stable process over a longer period, with a higher volume of tasks and clearer trends indicating consistent performance and effective error management.

## IV. DISCUSSION

The navigation took place in a warehouse with an area of 36 × 40 metres. During the testing, adequate changes in direction, obstacle avoidance and other deviations on the route were observed. A different route than the suggested method could be used, and after a while the navigation independently re-routed itself to the new route to the destination. From the results of the reliability measurements, it can be concluded that the method was relatively reliable and guided the user to the target to a high degree. It should not be forgotten that the method showed navigation line dropouts and is significantly sensitive to light conditions. This caused the user to become disoriented, especially against the light. The method is simple to understand and easy to use. Basically, you just ‘‘follow’’ the line. This feature needs to be tested for user acceptance on probands in future research to fully confirm its applicability in practice. Several virtual elements for notifications have been added to the application for ease of understanding and clarity. The method does not work with any safety features that could prevent or warn about a possible collision.

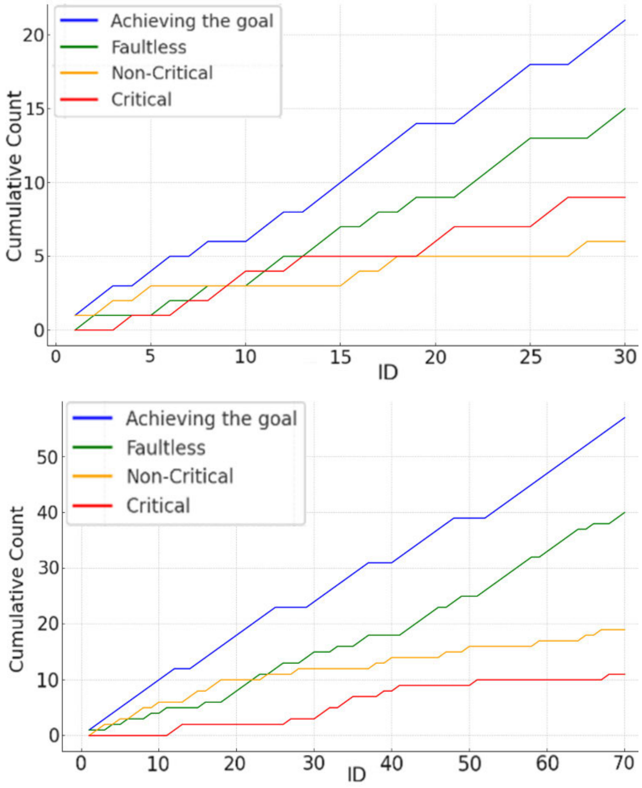


**FIGURE 10.** Cumulative flow diagram (cfd) for innovated application.

A similar approach was taken by Zhang et al. [22]. They created an application for the Android operating system in the Unity development environment using a combination of ARCore and AR Foundation, but compared GPS accuracy with the SLAM method - an inappropriate comparison for indoor navigation given the poor satellite signal. The waypoint accuracy of GPS is 0.52 metres, while the SLAM accuracy is substantially higher at 0.0059 metres. Ayyanchira et al. [31] created a similar system based on a 1:1 CAD baseline, which we also transferred to a development environment.

Liu and Meng [19] state that a significant contribution in AR navigation is creating virtual objects which also contain textual information. We have implemented this feature in a virtual-information panel that is displayed near the relevant point of interest - it adds information to the specific position and displays the name of the object. Huang et al. **[51]** work with beacons in contrast to the inertial sensors we applied in our study. This technology works on a similar principle to Bluetooth. Rubio-Sandoval et al. [67] also came up with a similar indoor system by applying position recognition using QR codes. The navigation line is based on Dijkstra's algorithm. The application is similar to that developed by [68], who only deal with the functionality of the application. Noreikis et al. [69] achieved less than 1 metre position accuracy and less than 5 degrees gaze direction accuracy for image-based k-localisation, with 90% of the target location achieved. The navigation is based on the A∗ algorithm and a 3D point cloud in a cloud environment. It is a hybrid approach that combines image-based localisation and inertial tracking. Sato et al. [70] use BLE Beacons for position localisation and achieve a navigation accuracy of 1.65 metres on average. Gang and Pyun [71] achieved an average accuracy of 1.61 metres when localising BLE Beacons combined with QR markers. Chidsin et al. [54] with a hybrid ORB-SLAM system achieved navigation with markers accurately 81.82% of the time and up to 99% of the time with navigation arrows. It can be said that even with a very accurate point cloud, the accuracy does not fully translate to augmented reality. Morar et al. [72] compared ARCore phone localisation technology with a VR solution, the HTC Vive device. The ARCore localisation technology exhibits lower average errors in trajectory and orientation compared to the HTC Vive solution. However, fluctuations in errors can still pose challenges. The limited lifespan of cloud anchor points remains a concern for applications intended for long-term deployment.

The proposed solution is seemingly quick and easy. It is also not necessary to use additional technologies in the form of Wi-Fi, RFID, etc. If it needs to be extended, it must be modified depending on the new conditions. This is also a limitation of the design, if it is an environment that is very changeable, then it needs to be modified for every change. So, the proposed method can be recommended for stable and long-term unchanging environments. In testing, there was an uncomfortable feeling of constantly looking at the display, the user has the feeling of moving in the display and not in reality, and there may also be a potential risk of collision with moving objects and people. From this it can be concluded that HMD glasses should be used as the primary navigation device for the AR system. During the measurements, the device exhibited increased temperature and touch sensitivity. A fully charged phone with a battery capacity of 4500 mAh discharged in less than 4 hours of testing. It is advisable to choose sufficiently powerful hardware with the latest version of Android. The application relies on the ARCore module to determine the location, which primarily uses image recognition, and the location is not modified in any way - it depends entirely on algorithms compiled by Google.

The Image Tracking feature was only tested in a trial version during application development and was not included in the final version of the application. The original idea of incorporating Image Tracking was to switch on the QR code reader when a marker was captured. From our tests it was found that Image Tracking was not unique enough for AR Foundation despite the embedded icons and distinctive frames for each position marker. When a new marker was inserted into the library, the algorithm did not recognise another marker.

## V. CONCLUSION

The main purpose of implementing this kind of application is to make it faster and easier to navigate in the warehouse, especially for people who are just starting their training. The current potential of this application is also its use for order picking and order preparation. Primarily, it works with a combination of ARCore and AR Foundation, which are designed for Android devices. During the design and development phases we also focused on the user interface and its usability. Optimisation and precision were key elements of our design. Therefore, each version was tested and then evaluated and improved.

Is it possible to achieve 99% accuracy in inertial sensor-based applications? Testing has confirmed that it is not possible in the current version. At most the target was reached 83% of the time, which is insufficient for practical applications. Higher accuracy could probably be achieved using a hybrid solution. A technology suitable for this is SLAM, which can potentially achieve these results, but it is necessary to implement a combined solution that would provide more accurate values. Again, there is a risk of increased inaccuracy as the conversion of the point cloud to augmented reality reduces accuracy.

Augmented reality is an indispensable technology for Industry 4.0. Although augmented reality is not currently being fully deployed in industry, this situation may change rapidly with increasing investment in hardware and software. Here we have focused on the logistics sector, where augmented reality applications appear to have the biggest potential and will form the backbone of Operator 4.0. In this area, augmented reality brings faster, more flexible, and clearer decision making. It makes it easier and helps to quickly identify the position and navigate to the target location. Augmented reality also reduces the risk of order complaints that are caused by human factors. It is necessary to highlight here an element missing in the current version, namely security. This element needs to be added and further improved and tested. Augmented reality holds great potential. With increasing interest and investment, this technology is improving, both in terms of hardware and software. We can see such rapid changes now with the large-scale emergence and application of AI-based systems. We can count on a similar trend in the future with augmented reality in many fields, not only in logistics.

The next direction of the application development can be seen in the improvement of position determination and to reduce the susceptibility of the application to lighting conditions. One possibility is to use Vuforia's Area Target technology. The risk here is that storage areas change frequently, or the environment is too homogeneous and difficult to recognise. A possible option is to choose a combination of technologies and incorporate radio technology such as UWB, RFID or SLAM into the design. It is essential to test the application on probands for usability and user-friendliness and process their suggestions for further development. It is also important from a usability perspective to test the application through HMD glasses for augmented reality, e.g. Microsoft HoloLens 2 or Apple Vision.

## REFERENCES

[1] E. Hofmann and M. Rüsch, ''Industry 4.0 and the current status as well as future prospects on logistics,'' *Comput. Ind.*, vol. 89, pp. 23–34, Aug. 2017, doi: 10.1016/j.compind.2017.04.002.

[2] H. Kagermann, W. Wahlster, and J. Helbig, ''Recommendations for implementing the strategic initiative INDUSTRIE 4.0,'' Industrie 4.0 Working Group, Nat. Acad. Sci. Eng., Washington, DC, USA, Tech. Rep. Accessed: Apr. 5, 2023. [Online]. Available: https://en.acatech.de/publication/recommendations-for-implementing-the-strategic-initiative-industrie-4-0-final-report-of-the-industrie-4-0-working-group/

[3] P. Wellener, J. Zale, and H. Ashton. *Pathways to Faster Innovation: Making Bold Moves in 'The Second Half of the Chessboard'*. Accessed: Apr. 6, 2023. [Online]. Available: https://www2.deloitte.com/content/dam/Deloitte/ec/Documents/energy-resources/DI_Pathways-to-faster-innovation

[4] S. Weyer, M. Schmitt, M. Ohmer, and D. Gorecky, ''Towards Industry 4.0—Standardization as the crucial challenge for highly modular, multi-vendor production systems,'' *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 579–584, Jan. 2015, doi: 10.1016/j.ifacol.2015.06.143.

[5] R. T. Azuma, ''A survey of augmented reality,'' *Presence, Teleoperators Virtual Environments*, vol. 6, no. 4, pp. 355–385, Aug. 1997, doi: 10.1162/pres.1997.6.4.355.

[6] H. Shoaib and S. W. Jaffry, ''A survey of augmented reality,'' in *Proc. Int. Conf. Virtual Augmented Reality (ICVAR)*, Monterrey, Mexico, Nov. 2015.

[7] R. Carey, T. Fields, A. van Dam, and D. Venolia, ''Why is 3-D interaction so hard and what can we really do about it?'' in *Proc. 21st Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*. New York, NY, USA: Association for Computing Machinery, Jun. 1994, pp. 492–493, doi: 10.1145/192161.192299.

[8] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, and B. MacIntyre, ''Recent advances in augmented reality,'' *IEEE Comput. Graph. Appl.*, vol. 21, no. 6, pp. 34–47, Dec. 2001, doi: 10.1109/38.963459.

[9] V. Kasapakis and D. Gavalas, ''Occlusion handling in outdoors augmented reality games,'' *Multimedia Tools Appl.*, vol. 76, no. 7, pp. 9829–9854, Apr. 2017, doi: 10.1007/s11042-016-3581-1.

[10] E. Karan, M. Morkoc, A. Şireci, and K. Fidanboylu, ''Indoor and outdoor navigation application development with augmented reality technology,'' in *Proc. 5th Int. Conf. Comput. Math. Eng. Sci.*, 2021, p. 220.

[11] P. Fraga-Lamas, T. M. FernáNdez-CarameS, Ó. Blanco-Novoa, and M. A. Vilar-Montesinos, ''A review on industrial augmented reality systems for the Industry 4.0 shipyard,'' *IEEE Access*, vol. 6, pp. 13358–13375, 2018, doi: 10.1109/ACCESS.2018.2808326.

[12] A. Ceruti, P. Marzocca, A. Liverani, and C. Bil, ''Maintenance in aeronautics in an Industry 4.0 context: The role of augmented reality and additive manufacturing,'' *J. Comput. Design Eng.*, vol. 6, no. 4, pp. 516–526, Oct. 2019, doi: 10.1016/j.jcde.2019.02.001.

[13] G. M. Santi, A. Ceruti, A. Liverani, and F. Osti, ''Augmented reality in Industry 4.0 and future innovation programs,'' *Technologies*, vol. 9, no. 2, p. 33, Apr. 2021, doi: 10.3390/technologies9020033.

[14] V. Reljić, I. Milenković, S. Dudić, J. Šulc, and B. Bajč i, ''Augmented reality applications in Industry 4.0 environment,'' *Appl. Sci.*, vol. 11, no. 12, p. 5592, Jun. 2021, doi: 10.3390/app11125592.

[15] T. Broum, P. Hořejší, M. Malaga, and P. Grzona, ''Competencies of industrial engineers for implementing augmented reality metadata systems,'' *Sustainability*, vol. 15, no. 1, p. 130, Dec. 2022, doi: 10.3390/su15010130.

[16] G. Gabajová, B. Furmannová, I. Medvecká, P. Grznár, M. Krajč ovič, and R. Furmann, ''Virtual training application by use of augmented and virtual reality under university technology enhanced learning in Slovakia,'' *Sustainability*, vol. 11, no. 23, p. 6677, Nov. 2019, doi: 10.3390/su11236677.

[17] *Google AR & VR*. Accessed: Jul. 17, 2023. [Online]. Available: https://arvr.google.com/

[18] R. L. M. Guarese and A. Maciel, ''Development and usability analysis of a mixed reality GPS navigation application for the Microsoft HoloLens,'' in *Advances in Computer Graphics* (Lecture Notes in Computer Science), vol. 11542, M. Gavrilova, J. Chang, N. M. Thalmann, E. Hitzer, and H. Ishikawa, Eds. Cham, Switzerland: Springer, 2019, pp. 431–437, doi: 10.1007/978-3-030-22514-8_41.

[19] B. Liu and L. Meng, ''Doctoral colloquium—Towards a better user interface of augmented reality based indoor navigation application,'' in *Proc. 6th Int. Conf. Immersive Learn. Res. Netw. (iLRN)*. San Luis Obispo, CA, USA: IEEE, Jun. 2020, pp. 392–394, doi: 10.23919/iLRN47897.2020.9155198.

[20] B. Li, D. Zou, D. Sartori, L. Pei, and W. Yu, ''TextSLAM: Visual SLAM with planar text features,'' in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*. Paris, France: IEEE, May 2020, pp. 2102–2108, doi: 10.1109/ICRA40945.2020.9197233.

[21] G. Gerstweiler, ''Guiding people in complex indoor environments using augmented reality,'' in *Proc. IEEE Conf. Virtual Reality 3D User Interfaces (VR)*. Reutlingen, Germany: IEEE, Mar. 2018, pp. 801–802, doi: 10.1109/VR.2018.8446138.

[22] H. Zhang, M. Guo, W. Zhao, J. Huang, Z. Meng, P. Lu, L. Sen, and B. Sheng, ''Visual indoor navigation using mobile augmented reality,'' in *Advances in Computer Graphics* (Lecture Notes in Computer Science), N. Magnenat-Thalmann, J. Zhang, J. Kim, G. Papagiannakis, B. Sheng, D. Thalmann, and M. Gavrilova, Eds. Cham, Switzerland: Springer, 2022, pp. 145–156, doi: 10.1007/978-3-031-23473-6_12.

[23] D. Gualda, M. C. Pérez-Rubio, J. Ureña, S. Pérez-Bachiller, J. M. Villadangos, Á. Hernández, J. J. García, and A. Jiménez, ''LOCATE-U.S.: Indoor positioning for mobile devices using encoded ultrasonic signals, inertial sensors and graph-matching,'' *Sensors*, vol. 21, no. 6, p. 1950, Mar. 2021, doi: 10.3390/s21061950.

[24] P. Trebuna, M. Mizerak, and J. Trojan, ''Establishing security measures for the protection of production workers through UWB real-time localization technology,'' *Acta Tecnología*, vol. 9, no. 2, pp. 39–43, Jun. 2023, doi: 10.22306/atec.v9i2.167.

[25] B. Szász, R. Fleiner, and A. Micsik, ''ILOC—Building in-door navigation services using linked data,'' in *Proc. 12th Joint Int. Conf. Semantic Syst. (SEMANTiCS), 1st Int. Workshop Semantic Change Evolving Semantics (SuCCESS)*, vol. 16952016, Leipzig, Germany, Sep. 2016, doi: 10.13140/RG.2.2.27418.64969.

[26] Z. F. Yount, S. J. Kass, and J. E. Arruda, ''Route learning with augmented reality navigation aids,'' *Transp. Res. F, Traffic Psychol. Behaviour*, vol. 88, pp. 132–140, Jul. 2022, doi: 10.1016/j.trf.2022.05.019.

[27] G. Bhorkar, ''A survey of augmented reality navigation,'' 2017, *arXiv:1708.05006*.

[28] P. Contreras, D. Chimbo, A. Tello, and M. Espinoza, ''Semantic web and augmented reality for searching people, events and points of interest within of a university campus,'' in *Proc. XLIII Latin Amer. Comput. Conf. (CLEI)*, Sep. 2017, pp. 1–10, doi: 10.1109/CLEI.2017.8226397.

[29] A. Sochenkova, N. Podzharaya, and K. Samouylov, ''Augmented reality application to emergency systems with usage of SLAM,'' in *Proc. 8th Medit. Conf. Embedded Comput. (MECO)*, Jun. 2019, pp. 1–4, doi: 10.1109/MECO.2019.8760019.

[30] Y. Gu, W. Chidsin, and I. Goncharenko, ''AR-based navigation using hybrid map,'' in *Proc. IEEE 3rd Global Conf. Life Sci. Technol. (LifeTech)*, Mar. 2021, pp. 266–267, doi: 10.1109/LifeTech52111.2021.9391866.

[31] A. Ayyanchira, E. Mahfoud, W. Wang, and A. Lu, ''Toward cross-platform immersive visualization for indoor navigation and collaboration with augmented reality,'' *J. Visualizat.*, vol. 25, no. 6, pp. 1249–1266, Dec. 2022, doi: 10.1007/s12650-022-00852-9.

[32] N. Pirzada, M. Y. Nayan, F. Subhan, M. F. Hassan, and M. A. Khan, ''Comparative analysis of active and passive indoor localization systems,'' *AASRI Procedia*, vol. 5, pp. 92–97, Jan. 2013, doi: 10.1016/j.aasri.2013.10.063.

[33] S. Carlin and K. Curran, ''An active low cost mesh networking indoor tracking system,'' *Int. J. Ambient Comput. Intell.*, vol. 6, no. 1, pp. 45–79, Jan. 2014, doi: 10.4018/ijaci.2014010104.

[34] J. Hong and T. Ohtsuki, ''Signal eigenvector-based device-free passive localization using array sensor,'' *IEEE Trans. Veh. Technol.*, vol. 64, no. 4, pp. 1354–1363, Apr. 2015, doi: 10.1109/TVT.2015.2397436.

[35] M. Seifeldin, A. Saeed, A. E. Kosba, A. El-Keyi, and M. Youssef, ''Nuzzer: A large-scale device-free passive localization system for wireless environments,'' *IEEE Trans. Mobile Comput.*, vol. 12, no. 7, pp. 1321–1334, Jul. 2013, doi: 10.1109/TMC.2012.106.

[36] K. Curran, ''Hybrid passive and active approach to tracking movement within indoor environments,'' *IET Commun.*, vol. 12, no. 10, pp. 1188–1194, Jun. 2018, doi: 10.1049/iet-com.2017.1099.

[37] A. S. Shewail, N. A. Elsayed, and H. H. Zayed, ''Survey of indoor tracking systems using augmented reality,'' *IAES Int. J. Artif. Intell. (IJ-AI)*, vol. 12, no. 1, p. 402, Mar. 2023, doi: 10.11591/ijai.v12.i1.pp402-414.

[38] W.-W. Kao and B. Q. Huy, ''Indoor navigation with smartphone-based visual SLAM and Bluetooth-connected wheel-robot,'' in *Proc. CACS Int. Autom. Control Conf. (CACS)*. Nantou, Taiwan: IEEE, Dec. 2013, pp. 395–400, doi: 10.1109/CACS.2013.6734167.

[39] G. Shi and Y. Ming, ''Survey of indoor positioning systems based on ultra-wideband (UWB) technology,'' in *Wireless Communications, Networking and Applications* (Lecture Notes in Electrical Engineering), vol. 348, Q.-A. Zeng, Ed. New Delhi, India: Springer, 2016, pp. 1269–1278, doi: 10.1007/978-81-322-2580-5_115.

[40] A. Arntz, D. Keßler, N. Borgert, N. Zengeler, M. Jansen, U. Handmann, and S. C. Eimler, ''Navigating a heavy industry environment using augmented reality—A comparison of two indoor navigation designs,'' in *Virtual, Augmented and Mixed Reality. Industrial and Everyday Life Applications* (Lecture Notes in Computer Science), J. Y. C. Chen and G. Fragomeni, Eds. Cham, Switzerland: Springer, 2020, pp. 3–18, doi: 10.1007/978-3-030-49698-2_1.

[41] C. Yoon, R. Louie, J. Ryan, M. Vu, H. Bang, W. Derksen, and P. Ruvolo, ''Leveraging augmented reality to create apps for people with visual disabilities: A case study in indoor navigation,'' in *Proc. 21st Int. ACM SIGACCESS Conf. Comput. Accessibility*. New York, NY, USA: Association for Computing Machinery, Oct. 2019, pp. 210–221, doi: 10.1145/3308561.3353788.

[42] D. Sarkar, B. Sato, and T. Murase, ''Algorithms for selecting higher wireless connection-capacity routes,'' in *Proc. IEEE 80th Veh. Technol. Conf. (VTC-Fall)*. Vancouver, BC, Canada: IEEE, Sep. 2014, pp. 1–6, doi: 10.1109/VTCFALL.2014.6965862.

[43] G. R. Jagadeesh and T. Srikanthan, ''Route computation in large road networks: A hierarchical approach,'' *IET Intell. Transp. Syst.*, vol. 2, no. 3, p. 219, 2008, doi: 10.1049/iet-its:20080012.

[44] M. Elgendy, M. Herperger, T. Guzsvinecz, and C. S. Lanyi, ''Indoor navigation for people with visual impairment using augmented reality markers,'' in *Proc. 10th IEEE Int. Conf. Cogn. Infocommunications (CogInfoCom)*, Oct. 2019, pp. 425–430, doi: 10.1109/CogInfoCom47531.2019.9089960.

[45] P. Verma, K. Agrawal, and V. Sarasvathi, ''Indoor navigation using augmented reality,'' in *Proc. 4th Int. Conf. Virtual Augmented Reality Simulations*. Sydney, NSW, Australia: ACM, Feb. 2020, pp. 58–63, doi: 10.1145/3385378.3385387.

[46] J. Keil, D. Edler, and F. Dickmann, ''Preparing the HoloLens for user studies: An augmented reality interface for the spatial adjustment of holographic objects in 3D indoor environments,'' *KN J. Cartography Geographic Inf.*, vol. 69, no. 3, pp. 205–215, Sep. 2019, doi: 10.1007/s42489-019-00025-z.

[47] X. H. Ng and W. N. Lim, ''Design of a mobile augmented reality-based indoor navigation system,'' in *Proc. 4th Int. Symp. Multidisciplinary Stud. Innov. Technol. (ISMSIT)*, Oct. 2020, pp. 1–6, doi: 10.1109/ISMSIT50672.2020.9255121.

[48] A. Corotan and J. J. Z. Irgen-Gioro, ''An indoor navigation robot using augmented reality,'' in *Proc. 5th Int. Conf. Control, Autom. Robot. (ICCAR)*. Beijing, China: IEEE, Apr. 2019, pp. 111–116, doi: 10.1109/ICCAR.2019.8813348.

[49] T. Michel, P. Genevès, H. Fourati, and N. Layaïda, ''Attitude estimation for indoor navigation and augmented reality with smartphones,'' *Pervas. Mobile Comput.*, vol. 46, pp. 96–121, Jun. 2018, doi: 10.1016/j.pmcj.2018.03.004.

[50] C. Real Ehrlich and J. Blankenbach, ''Indoor localization for pedestrians with real-time capability using multi-sensor smartphones,'' *Geo-spatial Inf. Sci.*, vol. 22, no. 2, pp. 73–88, Apr. 2019, doi: 10.1080/10095020.2019.1613778.

[51] B.-C. Huang, J. Hsu, E. T.-H. Chu, and H.-M. Wu, ''ARBIN: Augmented reality based indoor navigation system,'' *Sensors*, vol. 20, no. 20, p. 5890, Oct. 2020, doi: 10.3390/s20205890.

[52] V. Dosaya, S. Varshney, V. K. Parameshwarappa, A. Beniwal, and S. Tak, ''A low cost augmented reality system for wide area indoor navigation,'' in *Proc. Int. Conf. Decis. Aid Sci. Appl. (DASA)*, Nov. 2020, pp. 190–195, doi: 10.1109/DASA51403.2020.9317014.

[53] J. Yang, Z. Kang, L. Zeng, P. Hope Akwensi, and M. Sester, ''Semantics-guided reconstruction of indoor navigation elements from 3D colorized points,'' *ISPRS J. Photogramm. Remote Sens.*, vol. 173, pp. 238–261, Mar. 2021, doi: 10.1016/j.isprsjprs.2021.01.013.

[54] W. Chidsin, Y. Gu, and I. Goncharenko, ''AR-based navigation using RGB-D camera and hybrid map,'' *Sustainability*, vol. 13, no. 10, p. 5585, May 2021, doi: 10.3390/su13105585.

[55] F. Sato, ''Indoor navigation system based on augmented reality markers,'' in *Innovative Mobile and Internet Services in Ubiquitous Computing* (Advances in Intelligent Systems and Computing), L. Barolli and T. Enokido, Eds. Cham, Switzerland: Springer, 2018, pp. 266–274, doi: 10.1007/978-3-319-61542-4_25.

[56] Z. T. Çankiri, E. E. Marasli, S. Aktürk, S. Sonlu, and U. Güdükbay, ''Guido: Augmented reality for indoor navigation using commodity hardware,'' in *Proc. 24th Int. Conf. Inf. Visualisation (IV)*. Melbourne, VIC, Australia: IEEE, Sep. 2020, pp. 708–713, doi: 10.1109/IV51561.2020.00123.

[57] M. Orfanos, H. Perakis, V. Gikas, G. Retscher, T. Mpimis, I. Spyropoulou, and V. Papathanasopoulou, ''Testing and evaluation of Wi-Fi RTT ranging technology for personal mobility applications,'' *Sensors*, vol. 23, no. 5, p. 2829, Mar. 2023, doi: 10.3390/s23052829.

[58] G. Guo, R. Chen, F. Ye, Z. Liu, S. Xu, L. Huang, Z. Li, and L. Qian, ''A robust integration platform of Wi-Fi RTT, RSS signal, and MEMS-IMU for locating commercial smartphone indoors,'' *IEEE Internet Things J.*, vol. 9, no. 17, pp. 16322–16331, Sep. 2022, doi: 10.1109/JIOT.2022.3150958.

[59] *Unity Real-Time Development Platform | 3D, 2D, VR & AR Engine*. Accessed: Aug. 2, 2023. [Online]. Available: https://unity.com

[60] Google for Developers. *Getting Started With AR Foundation | ARCore*. Accessed: Aug. 2, 2023. [Online]. Available: https://developers.google.com/ar/develop/unity-arf/getting-started-ar-foundation

[61] *Build New Augmented Reality Experiences That Seamlessly Blend the Digital and Physical Worlds | ARCore | Google for Developers*. Accessed: Aug. 2, 2023. [Online]. Available: https://developers.google.com/ar

[62] *ZXing/ZXing'. ZXing Project*. Accessed: Aug. 2, 2023. [Online]. Available: https://github.com/zxing/zxing

[63] Apple Developer Documentation. *ARKit*. Accessed: Aug. 2, 2023. [Online]. Available: https://developer.apple.com/documentation/arkit/

[64] Google for Developers. *ARCore Supported Devices*. Accessed: Aug. 2, 2023. [Online]. Available: https://developers.google.com/ar/devices

[65] D. Tazartes, ''An historical perspective on inertial navigation systems,'' in *Proc. Int. Symp. Inertial Sensors Syst. (ISISS)*. Laguna Beach, CA, USA: IEEE, Feb. 2014, pp. 1–5, doi: 10.1109/ISISS.2014.6782505.

[66] D. Schmalstieg and G. Reitmayr, ''The world as a user interface: Augmented reality for ubiquitous computing,'' in *Location Based Services and TeleCartography* (Lecture Notes in Geoinformation and Cartography), G. Gartner, W. Cartwright, and M. P. Peterson, Eds. Berlin, Germany: Springer, 2007, pp. 369–391, doi: 10.1007/978-3-540-36728-4_28.

[67] J. I. Rubio-Sandoval, J. L. Martinez-Rodriguez, I. Lopez-Arevalo, A. B. Rios-Alvarado, A. J. Rodriguez-Rodriguez, and D. T. Vargas-Requena, ''An indoor navigation methodology for mobile devices by integrating augmented reality and semantic web,'' *Sensors*, vol. 21, no. 16, p. 5435, Aug. 2021, doi: 10.3390/s21165435.

[68] G. H. P. Kumar, N. Akhila, R. Aravind, and P. Mohith, ''Indoor navigation using AR technology,'' *Int. J. Eng. Appl. Sci. Technol.*, vol. 4, no. 9, pp. 356–359, Jan. 2020, doi: 10.33564/ijeast.2020.v04i09.045.

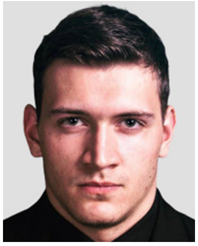
[69] M. Noreikis, Y. Xiao, and A. Ylä-Jääski, ''SeeNav: Seamless and energy-efficient indoor navigation using augmented reality,'' in *Proc. Thematic Workshops ACM Multimedia*. Mountain View, CA, USA: ACM, Oct. 2017, pp. 186–193, doi: 10.1145/3126686.3126733.

[70] D. Sato, U. Oh, J. Guerreiro, D. Ahmetovic, K. Naito, H. Takagi, K. M. Kitani, and C. Asakawa, ''NavCog3 in the wild: Large-scale blind indoor navigation assistant with semantic features,'' *ACM Trans. Accessible Comput.*, vol. 12, no. 3, pp. 1–30, Sep. 2019, doi: 10.1145/3340319.

[71] H.-S. Gang and J.-Y. Pyun, ''A smartphone indoor positioning system using hybrid localization technology,'' *Energies*, vol. 12, no. 19, p. 3702, Sep. 2019, doi: 10.3390/en12193702.

[72] A. Morar, M. A. Balutoiu, A. Moldoveanu, F. Moldoveanu, A. Butean, and V. Asavei, ''Evaluation of the ARCore indoor localization technology,'' in *Proc. 19th RoEduNet Conference: Netw. Educ. Res. (RoEduNet)*. Bucharest, Romania: IEEE, Dec. 2020, pp. 1–5, doi: 10.1109/RoEduNet51892.2020.9324849.

**PETR HOŘEJŠÍ** received the master's degree in industrial engineering from the Faculty of Mechanical Engineering, University of West Bohemia, Pilsen, Czech Republic, and the Ph.D. degree from the Department of Industrial Engineering and Management, University of West Bohemia. His Ph.D. thesis focused on Parallel and Distributed Simulation Based on High-Level Architecture Remote Management for Virtual Manufacturing Systems. He completed the Habilitation thesis with the title "Use of Virtual and Augmented Reality in Industrial Companies." Since 2006, he has been the author and an Assistant Professor with the Department of Industrial Engineering and Management, University of West Bohemia. In 2019, he was promoted to an Associate Professor of mechanical engineering. The teaching areas covered by him include technical informatics, computer support in mechanical engineering, digital factory and virtual reality, database systems in CIM, practice in computer technology, a methodological course (use of 3D data in archaeology), and virtual manufacturing systems of engineering enterprises (for Ph.D. degree). His experience abroad includes a long-term stay at the Universitá degli Studi di Genova, Italy, and recurring teaching stays at Manchester Metropolitan University, U.K., and Žilinská univerzita v Žiline, Slovakia. Recently, he has been most interested in the use of augmented and virtual reality in industry. He has been involved in several practical and commercial projects, such as implementing AR navigation for outdoor warehouses and using VR for production instructions synchronised with PLCs. He has also collaborated on multidisciplinary projects, including a VR Trainer for future teachers and VR Therapy for alcohol addiction.



**TOMÁŠ MACHÁČ** received the M.S. degree in industrial engineering and management from the University of West Bohemia, Pilsen, in 2022. He is currently pursuing the Ph.D. degree with the Department of Industrial Engineering and Management, University of West London. He was a Research Assistant with the Regional Technology Institute, which is a modern engineering and technological research centre at the Faculty of Mechanical Engineering, University of West Bohemia. His research interests include innovation applications of augmented reality to industrial environments, exploring the impact of technology on society and industry change, identifying, and advancing augmented reality for wider practical application, the impact of technology on user-friendliness and its acceptance. He is also interested in connectivity and communication in the use and support of augmented reality, metadata visualization, and logistics processes, specifically navigation support.

**MICHAL ŠIMON** is currently the Head of the Department of Industrial Engineering and Management, Faculty of Engineering, University of West Bohemia, Pilsen. He has long been involved in the intensive growth of the department, which is currently one of the finest and best-equipped industrial engineering departments in Czech Republic. His basic activities and areas of focus include-process rationalization, implementation of industrial engineering methods, automation and artificial intelligence in logistics processes, ergonomics, and the use of virtual reality. As a part of his long-standing project experience and implementation of methods and procedures in practice, he participated in the establishment of the first start-up company of the Faculty of Engineering, University of West Bohemia–Centre for Industrial Engineering Group (CIE), which is focused on the transfer of the above knowledge to business entities.

• • •