# Augmented Reality for Humans-Robots Interaction in Dynamic Slotting "Chaotic Storage" Smart Warehouses

7 authors, including:

Peter Papcun
Technical University of Kosice - Technicka univerzita v Kosiciach
67 PUBLICATIONS   1,843 CITATIONS

SEE PROFILE

Erik Kajáti
Technical University of Kosice - Technicka univerzita v Kosiciach
14 PUBLICATIONS   150 CITATIONS

SEE PROFILE

David Romero
Tecnológico de Monterrey
176 PUBLICATIONS   3,826 CITATIONS

SEE PROFILE

Lenka Landryová
VŠB-Technical University of Ostrava
60 PUBLICATIONS   91 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project    Intelligent human-centric lighting for mental wellbeing improvement View project

Project    Virtual Simulation and Training of Assembly and Service Processes in Digital Factories (VISTRA) View project

# Augmented Reality for Humans-Robots Interaction in Dynamic Slotting "Chaotic Storage" Smart Warehouses

Peter Papcun[1*], Jan Cabadaj[1], Erik Kajati[1], David Romero[2],
Lenka Landryova[3], Jan Vascak[1], Iveta Zolotova[1]

[1] Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering
and Informatics, Technical University of Kosice, Slovakia
peter.papcun@tuke.sk

[2] Tecnológico de Monterrey, Mexico
david.romero.diaz@gmail.com

[3] Department of Control Systems and Instrumentation, VSB – Technical University of Ostrava
Ostrava, Czech Republic
lenka.landryova@vsb.cz

**Abstract.** Nowadays, smart warehouses mostly use Automated Guided Vehicles (AGVs) controlled through magnetic or painted paths. This approach is suitable for "static slotting" warehouses, and for places where humans do not cross paths with mobile robots. Therefore, fixed-path AGVs are not an optimal solution for dynamic slotting "chaotic storage" warehouses, where picking and delivery paths are often changing. Hence, it is important to create an environment where AGVs have planned their path, and storekeepers can see their paths, and mark restricted areas by virtual means if needed, for these mobile robots and humans to move and stand safely around a smart warehouse. In this paper, we have proposed an Augmented Reality (AR) environment for storekeepers, where they can see an AGV planned path, and they can add virtual obstacles and walls to the mobile robots' cyber-physical navigation view. These virtual obstacles and walls can be used to determine restricted areas for mobile robots, which can be seen for example as safe areas for humans' and/or robots' stationary work. Finally, we introduce the system architecture supporting the proposed AR environment for humans-mobile robots safe and productive interaction.

**Keywords:** Industry 4.0, Smart Warehouses, Operator 4.0, Augmented Reality, Human-Robot Interaction, AGVs, Drones, Dynamic Slotting, Chaotic Storage.

## 1    Introduction

According to [1], in a "traditional warehouse", the operations of pickup, delivery, and bookkeeping are accomplished by (human) storekeepers. These manual activities have several drawbacks, particularly when it comes to storing and fetching inventories into and from bin rack storage systems, and keeping paper-based records of the stocks using account books, since both activities are (a) time-consuming, (b) prone to human-error, and (c) represent a waste of human talent – from a lean warehousing perspective [2]. Thus, it is imperative to make traditional warehouses *"smart",* meaning: 'automated,

unmanned, and paperless – when conducting the operations of pickup, delivery, and bookkeeping' [1]. Furthermore, as the retail revolution in e-commerce, and the rise of personalised products continue [3], the ability to manage a larger number of SKUs with fewer errors, and with faster pickup and delivery times (i.e. intra-logistics), have started to make "static slotting" (i.e. assigning a permanent location in a warehouse for each product [4]) not the best practice in the Industry 4.0 era. This considering the Industry 4.0 vision of "lot-size-one production" [5]. Therefore, "dynamic slotting" is emerging as the new best practice in "Warehousing 4.0" in order to accommodate the variability in products' turnover rates, and future wave of picking requirements for seasonal, mass-customized, and/or personalised products [4]. This "dynamic" approach will contribute to reducing time and transport, by having the location of the prime bins next to the shipping docks with the high moving products of e.g. the week [4]. Moreover, a more advanced and digitally-enabled dynamic slotting approach, known as "chaotic storage", is being practised by Amazon.com, and other online retailers, using any product empty bin with any random product that can fit into it, and having a software keeping track of the product location, and directing an Automated Guided Vehicle (AGV), drone or human to it via an optimised picking route [4]. To be able to control a movement of the AGV in a dynamic environment, two tasks must be solved at once: (i) determining the position of such AGV, and (ii) describing the environment where AGV is moving. Such an approach is well known as Simultaneous Localisation And Mapping (SLAM). It can be defined as a process that helps create a map of an unknown environment (including obstacles) along with its use to calculate the position of the AGV without any need of particular knowledge about its position [6]. SLAM encompasses a number of various methods like e.g. Graph SLAM [7], which utilizes the so-called sparse information matrices based on observation interdependencies. Meantime, further SLAM approaches were developed that differ by used sensors or the purpose of use. For instance, special SLAM for use of RGB cameras is presented in [8], which serves mainly in indoor applications. Another modification of SLAM for indoor tasks utilizes radio beacons and depth sensors (scanners) [9]. To complete this brief overview, in [10] an elderly care system is described, which is integrated together with SLAM.

In this paper, we introduce an Augmented Reality (AR) solution for Humans-Robots (i.e. AGVs and drones) safe and productive Interaction (HRI) in dynamic slotting "chaotic storage" smart warehouses, where these can communicate in a bidirectional way by means of AR mobile/wearable devices at humans, and computer vision at AGVs and drones. The main objective is to be able to visually share between humans, AGVs and drones their picking and delivery routes around the warehouse in order to avoid possible collisions and guarantee their safety.

## 2    AR & Computer Vision for Humans-Robots Interaction (HRI)

The research work and technological development described in this paper focuses on the possibilities of using AR mobile/wearable devices at humans (e.g. the Augmented & Collaborative Operator 4.0 [11]), and computer vision at mobile robots such as AGVs and drones (e.g. Human-Machine Interfaces (HMIs) 4.0 [12]), in the context of dynamic slotting "chaotic storage" smart warehouses. This in order to facilitate humans and mobile robots safe interaction and visual (control) communication. The advantage

of using *AR technology* as a shared and augmented visual language for communication between humans and mobile robots, in comparison to traditional and static visual controls (e.g. tracks made of an adhesive magnetic tape affixed or painted on the floor to be followed by AGVs, and respected by humans), is that it can cope with the dynamic nature of the "chaotic storage" smart warehouses, where picking and delivery routes will be often, almost always, changing for humans, AGVs, and drones. Therefore, in a cyber-physical smart warehouse environment, either real objects in the physical world or virtual objects specified by humans and/or mobile robots (e.g. virtual paths and virtual walls or fences), can be used to create safe picking and delivery routes, and restricted areas for the case of humans' and/or robots' stationary work, which mobile robots and humans will be able to distinguish. AR technology can provide additional data about the environment for the SLAM.

In the next sub-sections, we introduce the proposed AR-HRI system architecture design and provide the needed guidelines for its implementation.

## 2.1 Reference AR-HRI System Architecture Design

The AR-HRI system architecture design consists of three separate, but interrelated applications (apps): (i) a Core app installed in a desktop computer, and responsible for creating the "AR smart warehouse view" for the humans (i.e. the storekeepers) and for the mobile robots (i.e. the AGVs and drones); (ii) the AR app installed in a smart mobile phone, or in smart-glasses, allowing the humans, the storekeepers, to view any virtual object added in the smart warehouse cyber-physical space in their perception/view of the real-world; and (iii) the Robot app installed in an AGV or drone, which uses computer vision to also allow the mobile robots to have in this case a cyber-physical view of the smart warehouse as they move around it (see Fig. 1). These apps should support many different network protocols (e.g. Ethernet, Bluetooth, Radio Frequency technology) to be able to work with different robots or sensors. All these apps together, as an AR-HRI system, have the same goal, to provide humans and mobile robots with safe dynamic routes for picking and delivery products at the smart warehouse, avoiding any collision by means of AR visual controls (i.e. virtual paths and virtual walls).
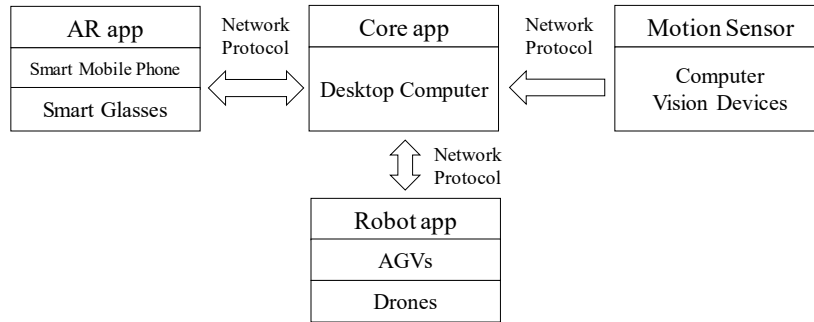


**Fig. 1.** Reference AR-HRI System Architecture Design

## 2.2 Particular AR-HRI System Architecture: Lab Platform Development

For developing and testing the proposed AR-HRI system architecture, the following software and hardware components where considered for the lab platform development: (i) for the Core app, C# programming language was used for the desktop app dev., since it is a common programming language among desktop and mobile apps development platforms, including Microsoft Kinect [13], and a desktop-computer with the following characteristics: 4-cores Intel i7-4710HQ CPU with base frequency at 2.50GHz, 16GB of RAM memory, GeForce GTX970M GPU, Windows 10 Pro OS; (ii) for the AR app, Unity development platform [14] was chosen due to its multiplatform support, and broad range of software development libraries and packages, which allow the possibly to extend the solution to other AR platforms such as Microsoft HoloLens [15], and a smart mobile phone with the following characteristics: 6-cores Snapdragon 650 w/ frequency at 1.80GHz, 3GB of RAM memory, Adreno 510 GPU, Android 6.0 OS; and (iii) for the Robot app, except for network communication, C programming language was used due to its popularity among robot programmers, and a Khepera III AGV [16], which runs an embedded Linux OS, and offers all standard C libraries for its programming (see Fig. 2).
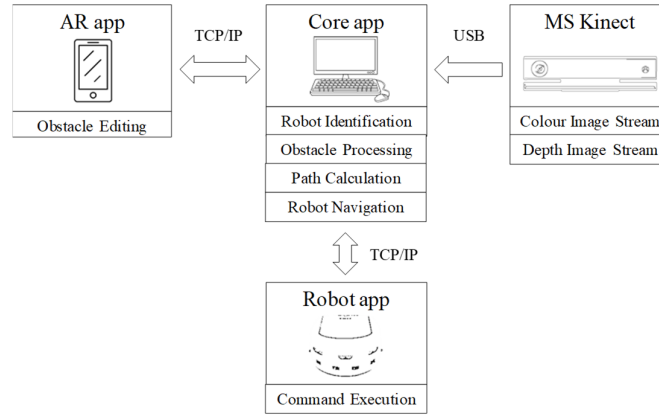


**Fig. 2.** Particular AR-HRI System Architecture Design: Lab Platform Development

Furthermore, the following lessons learned from the lab platform development are shared hereunder.

**AR app.** The first step to develop an AR app is to detect the scene, and register and differentiate the "virtual objects" in relation to the "physical ones". This itself is a broad topic, and many years of research have been already dedicated to it, so here it comes the advantage of using the Unity development platform [14] since it provides SDKs (Software Development Kits) focusing exactly on these problems. Developers usually divide AR-SDKs into two main categories based on their approach to scene-tracking: (i) marker-tracking tools, which are based on tracking a known pattern placed in the real-world and calculating its transform; and (ii) marker-less-tracking tools, which are designed to work in an unknown environment, usually by a combination of device

orientation, and location sensors comparing consecutive images from the camera. Thus, we have investigated and tried multiple AR-SDKs supporting marker-less-tracking for the lab platform development, namely: ARCore, Vuforia, Wikitude, Kudan, and Maxst. All these AR-SDKs were benchmarked based on their compatibility with peripheral devices, performance, quality of tracking, and license type. Maxst [17] was chosen at the end as the best for our AR platform solution since it performs well while using little computing resources. It is also compatible with most Android and iOS devices and offers a free license for non-commercial purposes as long as the application contains the watermark.

**Core app.** All the AR system logic is implemented in the Core app running on a desktop computer. It uses custom protocol built on TCP/IP sockets for efficient two-way connection with both the AR app and the Robot app. The first step is receiving data from the AR app, and additional data from the Microsoft Kinect [13] sensor. These are then combined and processed using various computer vision methods, resulting in the final map of the "cyber-physical smart warehouse environment" containing both virtual and physical objects. Next step is the calculation of the path/route that the mobile robot should take between its current and final position for a product pickup and delivery. Dijkstra's shortest path algorithm [18] was implemented for path planning. This algorithm was chosen because it guarantees to find the shortest path, and is relatively simple to implement. Its drawback is that the algorithm is not the most time-efficient one, but given the size of our pilot smart warehouse space, it only takes a few milliseconds to calculate the AGV path/route. The layout of the Core app can be seen in Fig. 3 and consists of four main parts.
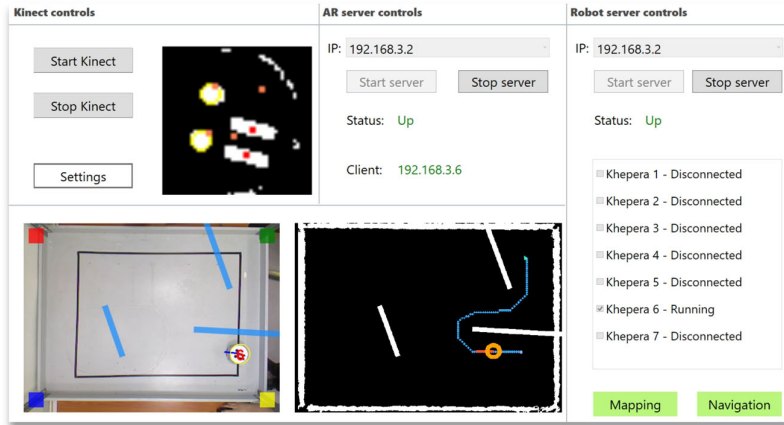


**Fig. 3.** The Core App Interface

The first three parts (i.e. Microsoft Kinect controls, AR server controls, and robot server controls) are used for communication setup with Microsoft Kinect sensor, AR app, and Robot app. The last part is used for the visualisation of the cyber-physical space. In the bottom-left part of Fig. 3, we offer a top view of the lab platform testing space with virtual obstacles (the blue rectangles); those recognized by the AGV (the

yellow circle); the blue line as the AGV path/route direction; and the red number (6) as the AGV identificator. The bottom-right part of Fig. 3 allows visualising the planned path for the AGV (the blue dotted line), the AGV actual direction (the red line), the AGVs position (the orange circle), and the virtual and physical obstacles (the white parts).

**Robot app.** This app serves as a client that connects to the Core app immediately after starting. Its only purpose is to set the speeds of the Khepera III AGV [16] motors based on commands incoming from the Core app and providing both infrared and ultrasonic sensor readings in case they are needed. Moreover, the Microsoft Kinect [13] sensor is used to help to solve two problems that appeared in the initial stage of the Robot app development: (i) determining the AGV's position and orientation (see Fig. 3, top-left section), and (ii) detecting physical obstacles in the real world.

### 2.3 Lab Platform Functionalities Testing

Thanks to the AR app developed, the user was able to view through the mobile device camera, the additional virtual walls inserted at the cyber-physical smart warehouse environment, and he/she was also able to add or remove these by simply drawing on the smart mobile phone screen. Furthermore, we have tested different placement, size, and shapes of the virtual obstacles. We have also tested how different obstacles affect the Dijkstra's shortest path planning algorithm [18]. We have proved that the path is achievable with no collisions for as long as the width of the whole calculated path from the start to the destination is at least 10% wider than the actual width of the AGV.

Moreover, the AR app and the Core app are connected through Wi-Fi, and each time the user makes changes, the Core app is notified of the changes made.

Fig. 4 shows an example scene. The three blue fences are the virtual walls drawn by the user.
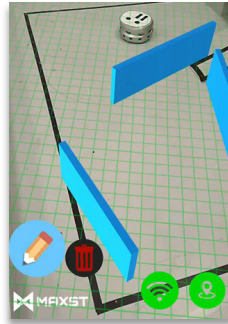


**Fig. 4.** Mobile Robot and Virtual Walls

The mobile robot was navigated following a virtual path. To achieve this, the Robot app was continually calculating the direction that the AGV should be moving on in order to keep following the path, then these values along with the mobile robot's current state are used as an input to the fuzzy system [19]. The navigation system is designed so that it outputs two values, one for each AGV wheel. These values are then sent to

the mobile robot using the same protocol (i.e. TCP/IP) as for communicating with the AR app. Additionally, the Khepera III AGV [16] has a marker placed on top of it. This marker contains several simple shapes arranged in a specified order. Since the layout of the shapes is known (see Fig. 4), we can use it to find the mobile robot in the image and determine its location and orientation. The image is obtained from the Microsoft Kinect colour camera. This task can be also solved using only images from the AR app, but since the smart mobile phone camera is not stationary, it would be significantly more complicated, and the user would be required to keep the mobile robot in the camera view all the time. Hence, the problem of detecting physical objects would be even harder because images in the AR app are captured using only one camera, which means that they lack any information about the depth of the scene. There are techniques for estimating depth from a single image, but these are not accurate enough for our case. The Microsoft Kinect depth sensor has proven to be a good choice thanks to its high precision.

## 3  Conclusions and Future Research

Although not yet widely adopted, AGVs and drones will revolutionise the way products are transported inside a smart warehouse. Nevertheless, not all smart warehouses will become "unmanned", some may still require humans, AGVs and drones collaboration for their optimal performance, so it is in these cases where our AR solution for humans-mobile robots safe and productive interaction aims to contribute. Based on the literature review conducted, we have created a reference system architecture that describes an AR implementation into traditional SLAM techniques. Moreover, we have developed a case study based on the proposed architecture to perform functionalities testing by observations.

Further research will focus on testing other AR wearable/mobile devices such as smart-glasses, testing other shortest path planning algorithms, and testing other robots' localization techniques. Thus, using AR wearable devices such as Microsoft HoloLens headset [15] will bring the advantage of hands-free to storekeepers. Algorithms able to solve path search and planning under "dynamic changes" of warehouse parameters promise considerable improvements to our current solution. Particularly, modifications of the D* algorithm, which is an extension of the Dijkstra's path planning algorithm, will be in the focus of future research [20] [21]. We have used a Microsoft Kinect sensor for localization of the AGVs in lab testing space. In future research, we also plan to focus on more suitable ways of AGVs localization in large warehouses, like Wireless Node Monte Carlo Localization [22], positioning based on RFID technology [23], or localization based on Bluetooth Low Energy [24].

# References

1. Liu, X., Cao, J., Yang, Y., Jiang, S.: CPS-based Smart Warehouse for Industry 4.0: A Survey of the Underlying Technologies. Computers, 7(1):1-13 (2018)
2. Ackerman, K.: Lean Warehousing. Ackerman Publications (2017)
3. Shukla, P.S., Nigam, P.V.: E-Shopping using Mobile Apps and the Emerging Consumer in the Digital Age of Retail Hyper Personalization: An Insight. Pacific Business Review International, 10(10):131-139 (2018)
4. MicroChannel: Warehouse Management: Static vs. Dynamic Slotting. Blog Post (2015) <http://microchannel.com.au/ company-blog/march-2015/warehouse-management-static-vs-dynamic-slotting>
5. Zuehlke, D.: Smart Factory – Towards a Factory-of-Things. Annual Reviews in Control, 34(1):129-138 (2010)
6. H. Durrant-Whyte, T. Bailey: Simultaneous Localization And Mapping: Part I. IEEE Robotics & Automation Magazine, 13(2):99-110 (2006)
7. S. Thrun, M. Montemerlo: The Graph SLAM Algorithm with Applications to Large-Scale Mapping of Urban Structures, International Journal of Robotics Research, Sage Publications Inc., Vol. 25, pp. 403-429 (2006)
8. Tian, R., Zhang, Y., Gao, C., Deng, Y., Jiang, H.: Semi-Direct RGB-D Slam Algorithm for Mobile Robot in Dynamic Indoor Environments, Repository of Belarusian National Technical University (2018)
9. J. Vaščák, I. Savko: Radio Beacons in Indoor Navigation. World Symposium on Digital Intelligence for Systems and Machines, pp. 283-288 (2018)
10. J.Y. Chen, L. Huang: Design of Elderly Care System Integrated with SLAM Algorithm. International Conference on Security with Intelligent Computing and Big-data Services, Springer, pp. 503-513 (2018)
11. Romero, D., Stahre, J., Wuest, T., Noran, O., Bernus, P., Fast-Berglund, Å., Gorecky, D.: Towards an Operator 4.0 Typology: A Human-Centric Perspective on the Fourth Industrial Revolution Technologies. International Conference on Computers & Industrial Engineering (CIE46) Proceedings, Tianjin/China, pp. 1-11 (2016)
12. Papcun, P., Kajati, E., Koziorek, J.: Human Machine Interface in Concept of Industry 4.0. World Symposium on Digital Intelligence for Systems and Machines, pp. 289-296 (2018)
13. Zhang, Z.: Microsoft Kinect Sensor and its Effect. IEEE MultiMedia, 19(2):4-10 (2012)
14. Unity Technologies: Unity <https://unity3d.com/>
15. Microsoft HoloLens <https://www.microsoft.com/en-us/hololens/>
16. Khepera III <https://www.k-team.com/mobile-robotics-products/old-products/khepera-iii>
17. Maxst <http://maxst.com/#/>
18. Dijkstra, W.E.: A Note on Two Problems in Connexion With Graphs. Numerische Mathematik 1, pp. 269-271 (1959)
19. Zadeh, L.A.: Fuzzy Sets. Information and Control, Vol 8, pp. 338-353 (1965)
20. Koenig, S., Likhachev, M.: Fast Replanning for Navigation in Unknown Terrain. IEEE Transactions on Robotics, 21(3):354-363 (2005)
21. Sun, B., Zhu, D.: Three Dimensional D*Lite Path Planning for Autonomous Underwater Vehicle under Partly Unknown Environment. 12th World Congress on Intelligent Control and Automation, pp. 3248-3252 (2016)
22. Kurecka, A., Konecny, J., Prauzek, M., Koziorek, J.: Monte Carlo Based Wireless Node Localization. Elektronika ir Elektrotechnika, 20(6):12-16 (2014)
23. Lu, S., Xu, C., Zhong, R. Y., et al.: A RFID-enabled Positioning System in Automated Guided Vehicle for Smart Factories. Journal Of Manufacturing Systems, 44:179-190 (2017)
24. Kajati, E.: Edge-enabled Approach for Intelligent Human-System Interoperability. 18th Scientific Conference of Young Researchers (SCYR), (accepted, not yet published) (2019)