Towards a Remote Warehouse Management System

Ioannis Stamelos

Aristotle University of Thessaloniki, Greece
stamelos@csd.auth.gr

Charalampos Avratoglou

Entersoft, Greece
avr@entersoft.gr

Panayotis Tzinis

Aristotle University of Thessaloniki, Greece
panayotis.tzinis@gmail.com

George Kakarontzas

University of Thessaly, Greece
gkakaron@teilar.gr

Alexander Chatzigeorgiou

University of Macedonia, Greece
achat@uom.gr

Apostolos Ampatzoglou

Aristotle University of Thessaloniki, Greece
apamp@csd.auth.gr

Dimitris Folinas

International Hellenic University, Greece

dfolinas@gmail.com

Iakovos Stratigakis

Aristotle University of Thessaloniki, Greece iakovosds@csd.auth.gr

Lambros Karavidas

Aristotle University of Thessaloniki, Greece
karavidas@csd.auth.gr

Christina Volioti

Aristotle University of Thessaloniki, Greece
chvolioti@gmail.com

Theodoros Amanatidis

University of Macedonia, Greece
tamanatidis@uom.edu.gr

Anastasia Deliga

Aristotle University of Thessaloniki, Greece
adeliga@csd.auth.gr

Thrasyvoulos Tsiatsos

Aristotle University of Thessaloniki, Greece
tsiatsos@csd.auth.gr

Abstract. This paper describes the software architecture and the required hardware components of a Remote Warehouse Management System using Augmented Reality, designed in the context of the project called WMS & AR project. The system architecture will augment the capabilities of an existing WMS system [1]. The main services that the extended system will provide are (a) automatic recording of the layout of the warehouse; (b) automatic measurement of goods; (c) exception handling. Moreover, the paper discusses the architectural requirements of the system, presents the proposed software architecture and refers to the future steps.

Keywords: Warehouse Management System, System Architecture, Augmented Reality

1 Introduction

A warehouse management system (WMS) is software and processes that help the organizations with the warehouse operations that should be conducted so as the warehouse to be fully functional. These operations start from the moment that materials or goods enter a warehouse until the move out. In addition to the basic core functions, other ones more extensive are integrates to WMS to check the system states in order to have data to optimize the organization's strategy.
Warehouse management systems come in a variety of types. It can more or less complex and it should be adjustable to the size of the business. Some organizations build the own WMS from scratch. However, it is mostly seen businesses to implement a WMS from an established vendor.
A WMS system typically integrates the following subsystems

- An integration with an ERP, to accept requests and external events that affect the Warehouse
- The main WMS UI for back-office users to issue and plan WH actions, monitor progress, measure performance of operations etc.
- A set of mobile (typically RF) devices, held by WH operators, to issue and monitor proper W/H job order execution
- A set of touch stations to support assembly, packing or unpacking operations
- Several automation devices such as beacons, lights etc. used to automate WH processes.

In summary, it can be said that a warehouse management system (WMS) is a complex software structure with many integrated functionalities. It usually receives order from the overlying host system and manages them with the aim to be as optimized as possible in order to store as few goods as possible, using few resources and personnel for storage and delivery to the end customer.

2 Short presentation of WMS & AR project

The WMS+AR system aims to exploit the benefits of AR for a Warehouse Management System. The AR subsystem is envisaged to undertake the responsibility of identifying and rendering the Warehouse and its contents and provide a means to map and validate the above against the WMS logical information.

Thus, the AR system shall link to the logical entities of the WMS, such as Warehouses, Storage locations, and packages. It shall monitor the Warehouse contents, and it shall also provide near real-time rendering based on the identified 3D Model of the Warehouse at the Storage Location level and the photo images taken during the operation.

The AR system is envisaged to be used as a straightforward and easy to use means to automate stock counting tasks in a relatively low cost compared to other methods. Even though there are several methods to accomplish this, they either involve manual counting or infrastructure which has relatively high cost of ownership & operations such as RF IDs etc.

The AR system is expected to run continuously to pinpoint exceptions or discrepancies from the logical information maintained by the WMS.

The WMS user experience is expected to be substantially enhanced by 3D rending of the Warehouse while the user is expected to have increased control due to contextual image rendering, at the Storage location level, and the ability for the user to issue stock counting actions, either ad-hoc or automatically, based on predefined tactics, such as follow-the-picker, which shall revalidate the contents of the storage locations after a change is registered in the WMS.

Additional benefits provided by the AR system will be to initially map the physical coordinates to the Warehouse Storage locations and using AR data to inform or navigate users.

3 Technical issues of the WMS & AR System

This section refers briefly to the basic technical issues of the system. The proposed system should do at least three basic functions:

- The automatic recording of the layout of the warehouse.
- The automatic measuring of the goods that are always in the warehouse.
- Management of exceptions, where the term "exception" refers to any discrepancy between the automatic measurements and the stored measurements in the Warehouse Management System (WMS) database.

These issues are presented in the following paragraphs

3.1 Automatic recording of the layout of the warehouse

The layout of the warehouse must be recorded automatically and can be done either in a warehouse that is completely empty but also in warehouses with merchandise. This implies three basic technical capabilities:

- The capability of navigating the warehouse in some way. The warehouse can be navigated: by:
  - An employee who holds a camera or mobile phone with suitable software.
  - A robot that moves through the corridors of the warehouse and carries a camera or mobile with suitable software on a telescopic arm.
  - A drone which moves to the premises of the warehouse and carries a camera or mobile phone with suitable software.
  - A camera attached to a fixed-orbit mechanism.
- The capability to identify the storage locations of the warehouse given the storage system. Possible storage systems are (a) Back-to-back (b) Stack and (c) Drive-in (see Figure 1). In this project we will focus on warehouses with back-to-back sys-

tems. The identification can be carried out by a trained neural network depending on the type of warehouse and can be done

- On the device from (1) with the appropriate software, or
- In a middleware server on which Video recorded on the device (1) is transmitted.

1. The capability to measure the dimensions of storage spaces (length, width, height). This feature involves cameras suitable for capturing accurately these dimensions (e.g. special telephones or cameras).

Fig. 1. Storage systems

Finally, in Entersoft's WMS system there should be a spatial representation as shown in Figure 2:

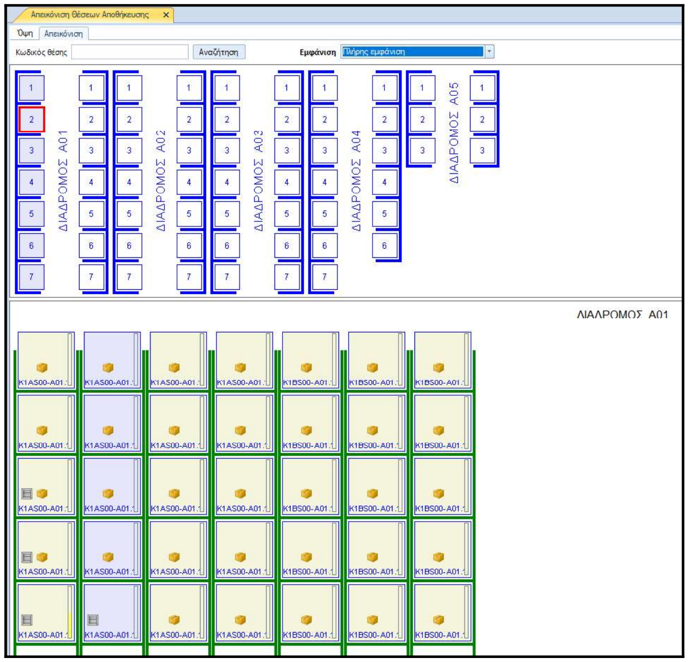

Fig. 1. Warehouse layout in Entersoft WMS

3.2 Automatic measurement of goods

Every kind of merchandise in the warehouse has various characteristics, which are stored in Entersoft WMS, as Code, Description and Barcode (see Figure 3).

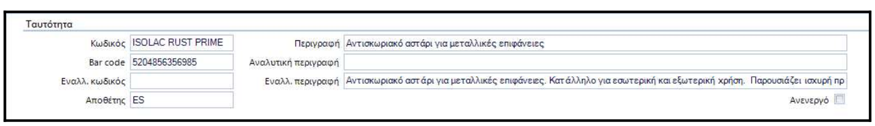

Fig. 1. Merchandise kind description in Entersoft WMS

The system should recognize the barcode of each product, which is packaged in the warehouse and should automatically count the goods. The measurement should also take into account the characteristics of the packaging:

- Measurement unit: which can be KIB (crates), ΚΙΛ (kg) or TEM (pieces), while there may be a basic unit of measurement of the packaging (e.g., TEM) and secondary unit of measurement (for example, pieces have weight).
- Volumetric characteristics: Height, length, width (in centimeters), area and volume
- Weight: net and gross

Also related to the counting of products, is the concept of the container. The container may be sacks, barrels or boxes, as shown in Figure 4.

Fig. 1. Containers

Therefore, the system should be capable in recognizing:

- Barcodes of storage locations and products
- Containers and
- Packagings

Considering the above, the system should be capable to measure the goods in the warehouse.

3.3 Exception Handling

The term "exception" refers to differences between the observed state in the warehouse by the devices and the recorded state in the WMS database.

The exceptions that will be handled by the system include at least the following:

- Quantity of stock,
- Location of stock,
- Constraints of placement regarding the various stored items.
- And possibly other conditions,

The handling of the exceptions will begin when there is a discrepancy between the state of the warehouse recorded in the WMS database and the state observed from the WMS&AR system.

In the proposed architecture several software components will provide the role of "middleware" between the WMS devices and the WMS system. These components will be executed in the WMS&AR Services server in the warehouse premises They will be responsible for (a) recording the observed values and sending them back to the WMS and (b) receiving actions from the WMS (e.g. an action would be a request that an employee should move to a specific WMS location for handling a possible exception etc.)

4 Architectural Requirements

4.1 Stakeholder concerns

Both Entersoft and the users of the system (i.e. companies that use the WMS system) require an economically efficient solution that improves the current status concerning the accuracy of the reported state of the warehouse. In addition, a solution that can be integrated easily with the current WMS system is required.

4.2 Constraints

The proposed solution should be economically not very expensive and no other special hardware or software should be required. The solution should be usable in both empty and filled warehouses (e.g. we do not assume that the warehouse is currently empty for the first automated layout recognition of the warehouse by the system). Moreover, the proposed architecture of the WMS&AR Services middleware should be interoperable with the Warehouse Management System.

4.3 Non – Functional Requirements

The most important non-functional requirements are the following:

1. Energy savings and recharging of the motion device (e.g. drone). The motion device should be capable of some autonomy before recharging and it should be possible to work efficiently. To this end, our solution is the handheld device to be able to carry the same actions as the motion device. Hence, when the motion device is charging on a docking point in the warehouse, an urgent request can be handled by the handheld device. Lastly, the motion device will carry out certain actions in periodic intervals and others non-periodic when available and told to do so by the Manager component of the WMS & AR Services middleware.
2. Speed of calculations for the volume measurements, barcode recognition and other possibly intensive algorithms. These measurements in the proposed architecture are carried out on the WMS&AR Services middleware server, which will be located in the warehouse premises in order to not be carried out by the devices. There might also be some exceptions to this, such as real – time recognition of objects and real – time object measuring.
3. Energy efficiency for the motion and handheld devices. The motion devices should be used in such manner so that energy is preserved. In the proposed architecture the majority of the intensive calculations will be carried out in the WMS&AR Services middleware server so that energy of these devices in not unnecessarily wasted.

4.4 Project risks

The major risk of the project is that the WMS&AR Services middleware will not be able to provide accurate measurements of the warehouse for the purposes of identifying exceptions. To this end some intensive prototyping already has been done and we have sufficient evidence that the proposed architecture will be able to address the risk.

5 Proposed Software Architecture

This section describes the software architecture and the required hardware components of the WMS & AR (Warehouse Management System and Augmented Reality). It follows largely the software documentation template proposed by Gorton in [8]. In Figure 5 the high-level overview of the architecture is depicted. The system under development (SuD) is communicating with the existing WMS.
The main concern of this document is the Augmented Reality Warehouse Management Services (AR WMS Services) components, which comprise components that are executed in the premises of the warehouse. The reason that this document will not be concerned further with the current WMS, it that the WMS software architecture will not change but only additions to the existing WMS system will be made for integrating the new functionality of the AR WMS Services components.

The main components of the AR WMS Services are the following:

1. **High Level - AR Actions API**: This component will provide an API that will allow calls from the WMS server to give instructions to the AR WMS Services subsystem. Actions can include instructions for fetching various pieces of information (e.g. volume of stock on a shelf etc.), but also sets of instructions (e.g. a set of measurements).
4. **Visualization Server:** This server will provide a visualization of the warehouse and specific shelves at a given time which will be accessed by the WMS Windows client. It provides access to the various .obj model files stored by the Visual Processor component at a specific storage location.
5. **Manager:** This will be the core component of the AR WMS Services system coordinating all the other components. Its purpose is to forward messages between the components. For example, it will forward incoming actions to the motion controller and also to the Tango device depending on the system configuration. It will also schedule incoming actions via a queue to keep track of the pending, currently operating, completed and failed actions. This component will be also responsible for marshalling and transferring the shelf measurements (barcodes, volumes etc.) to the WMS using the existing WMS WebAPI. Notice that this component is also responsible for making micro-corrections in the data sent back to the WMS system. For example, the same shelf can be read many times by the warehouse devices and small variations in the computed volume can occur. This subsystem will apply a policy to correct such variations (e.g. by averaging the various measurements).

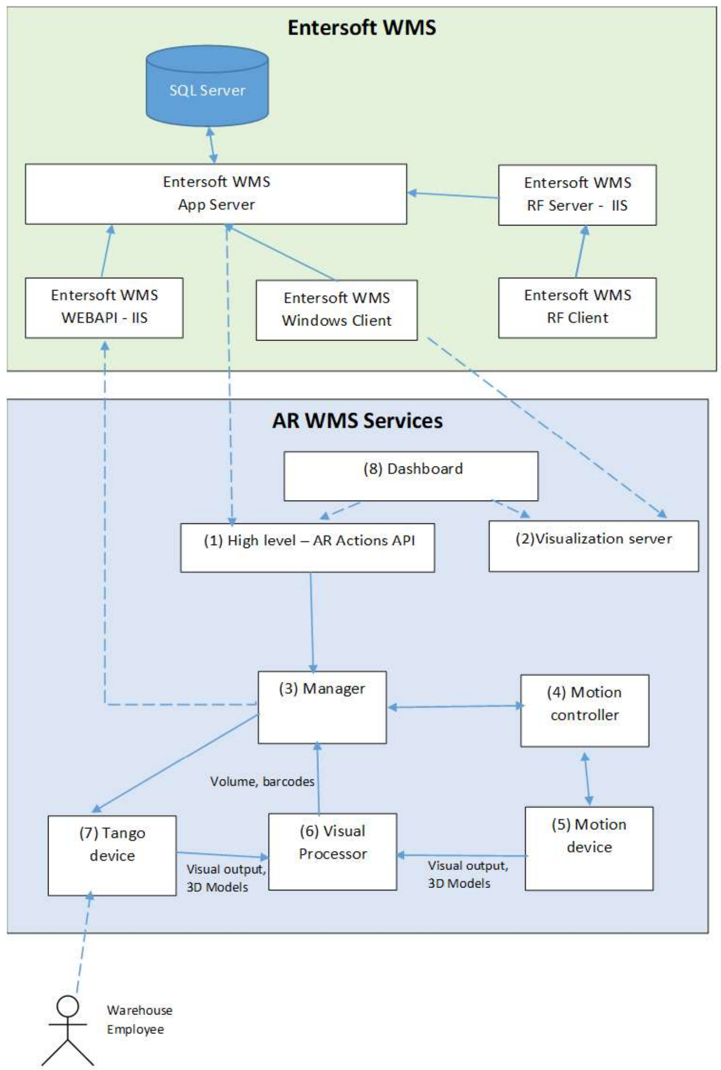

Fig. 1. High-level architecture overview

6. **Motion Controller:** This is the subsystem that controls the motion device (e.g. the Drone). For example, it instructs the drone to start a scanning session or return to its docking position or it may instruct the drone to go and capture visually a specific shelf. This component is also responsible for calculating the route that the motion device will follow in order to go to a specific point in the warehouse. The route will be the result of employed shortest-path algorithms.

7. **Motion Device:** A device moving in the warehouse autonomously (e.g. a Drone) according to the instructions given by the Motion Controller. Basically, it fulfills the same purposes as the Tango Device: it scans the warehouse and/or specific selves, generates 3D models and sends them to the Visual Processor for further computation.
8. **Tango Device**: A device that runs Google Tango (e.g. Tango Smartphone) and is operated by a Warehouse Employee. Basically, it fulfills the same purposes as the Motion Device: it scans the warehouse and returns the 3D models of the warehouse and/or a specific shelf. Then it sends the output to the Visual Processor for further computation.
9. **Visual Processor**: This component receives the visual output and the 3d models from the devices. It saves the .obj model files at a specific path from which the Visualization Server can access them. Through image processing, it is possible to calculate the volume of the captured 3d model and read the barcodes from the image. Notice that volume calculation can also take place at the Visualization Server component and currently we are assessing the optimal place for this service from a technology appropriateness point of view since the Visualization Server is developed using the Unity framework and the Visual Processor is using various programming languages and frameworks including Python and C. When completed the volume and the barcodes are sent to the Manager. The above process could also be implemented in the devices; however, this is an energy-consuming process that would overload the resources of the devices. This is the main reason why it was decided to be implemented in the WMS&AR Services middleware and not in the devices.
10. **Dashboard**: The Dashboard is going to communicate with (1) the High-Level AR Actions API (ordering of actions, listing of actions, cancellation of actions, listing of WMS-AR exceptions) and (2) the Visualization Server, via a REST API over HTTP(S).

6 Conclusions and next steps

In conclusion we are confident that the proposed architecture is solid enough and ready to be a reality. Thinking towards this direction our next steps are to find hardware and software that fit our needs and acquire the most appropriate one. A state of the art analysis was conducted to figure out our options.

Considering what was analyzed previously and the needs of a WMS AR system, it is obvious that our system should be able to record the layout of the warehouse. To do so the following solutions were found to be the most suitable one.

- **Matterport Pro2 3D Camera**: As found in the company's official site [2], the camera has a built-in Wi-Fi sensor in order to be paired with an iOS device, using it as a controller. With an intuitive interface the user can produce a 3D model of a space. The camera is equipped with an infrared 3D sensor and is 99% accurate. However, it must stand on a tripod to scan an area and must be paired with an iOS device so as to be controlled and return the data through the Capture app.

- **Structure sensor by Occipital:** A sensor that can be attached to the camera of an iOS device [3] and produce a 3D map or even a model of an object or an area. It is a light (95g) sensor and can be used for 3-4 hours straight of active sensing or can be even charged throughout its use. The camera itself uses an infrared structured light projector. The operating temperature is between 0 and 35 Celsius degrees.
- **Google Tango supported device:** Combined with a motion tracking camera, a 3D depth sensor, an accelerometer, a barometer, a gyroscope and a GPS, a Tango – compatible device, the ability not only to detect their position relative to their world but also construct a 3D map of their environment.

Moreover, to achieve the automated measurement of the commodities inside the warehouse an autonomous or semi-autonomous machine should be used to go through the corridors of the warehouse with the ability to find a specific location. By thinking towards that direction mobile robots either on the ground or drones match our needs.

A mobile robot is a robot that is capable of locomotion and is not fixed to one physical area. Mobile robots can be "autonomous" (AMR - autonomous mobile robot) or depend on direction gadgets (AGV - autonomous guided vehicle). Some of our solutions are the following:

- **KMR iiwa by KUKA:** A location – independent and highly flexible lightweight robot [4]. Equipped with seven special joint torque sensors on each axis that make it exceedingly sensitive to its environment. It screens its environment and responds quickly if an individual or item is standing out. It can move omnidirectionally and its positioning accuracy is up to +/-5millimeter. Lastly it is independent as the vehicle and robot are supplied directly with power from Li-ion batteries.
- **Gapter Drone:** Gapter is an unprecedented drone [5] that supports Robot Operating System (ROS), the MAVLink protocol, and the mavros wrapper layer between ROS and MAVLink. Gapter can work both indoor and outdoor with its inbuilt sensors like GPS and optical flow sensors and allows can be extended by add new modules or sensors. The Ordroid XU4 single-board computer integrated to provide the best performance possible for the autopilot and combined with a Rplidar laser scan it can build a map of the space and navigate within it.

In the case that an AGV is used some directional gadgets should be used in order to make it aware of its surroundings and make it possible for it to move with no collisions.

- **Dragonfly:** Dragonfly is a visual 3D positioning/location system [6] based on Visual SLAM. Visual SLAM, otherwise called vSLAM, is an innovation ready to fabricate a map of an obscure domain and guide, at the same time utilizing the mostly manufactured map, utilizing just computer vision. The main sensor required is a camera that must be mounted on board of the gadget. No other outside sensors are required.
- **R2000 2-D-LiDAR**: R2000 2-D-LiDAR is a photoelectric sensor [7] that monitors the AGV's surroundings with a 360° angle and provides measurements to the vehicle controller. Employees' legs can be filtered out of the high-resolution data and

utilized as a source of perspective point to distinguish their course of movement and follow them closely.

Trying to fulfill the needs and restrictions of our project we will proceed into the making of our architecture by using a Google Tango-compatible device and a Gapter Drone. To start with, both seem economically not so expensive as the other "contestants". In addition, the Tango-compatible device is preferred to the other options as it can be carried around without a tripod and has no room temperature limitations. Lastly, the Gapter drone is so powerful that can navigate on its own and versatile enough to operate indoors, such as warehouse.

7 Acknowledgment

This research has been co-financed by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH – CREATE – INNOVATE (project code: T1EDK-03502).

References

1. Entersoft WMS product page, https://www.entersoft.gr/products/wms/, last accessed 2019/07/04
2. Matterport Pro2 Camera page, htps://matterport.com/pro2-3d-camera/, last accessed 2019/07/04
3. Structure sensor page, https://structure.io/structure-sensor, last accessed 2019/07/04
2. KMR iiwa page, https://www.kuka.com/en-my/products/mobility/mobile-robots/kmr-iiwa, last accessed 2019/07/04
3. Gapter overview page, http://edu.gaitech.hk/gapter/overview-and-features.html, last accessed 2019/07/04
4. Dragonfly Homepage, https://www.dragonflycv.com/, last accessed 2019/07/04
5. Pepperl + Fuchs Sensors for AVG Vehicles page, https://www.pepperl-fuchs.com/global/en/32763.htm, last accessed 2019/07/04
6. Gorton Ian, Essential Software Architecture 2nd ed., Springer, 2011

View publication stats