LARA: a location-based and augmented reality assistive system for underground utilities' networks through GNSS

E. Stylianidis
H2020 LARA project coordinator
Email: stratos@geoimaging.com.cy

E. Valari, K. Smagas
GeoImaging Ltd, Nicosia, Cyprus
Email: elena,kostas@geoimaging.com.cy

A. Pagani, J. Henriques
German Research Center for Artificial Intelligence (DFKI)
Email: alain.pagani@dfki.de, jhenriques@dfki.uni-kl.de

A. Garca, E. Jimeno, I. Carrillo
Ingeniera y Soluciones Informticas S.L., Seville, Spain
Email: agarcia,elisa.jimeno,ivan.carrillo@isoin.es

P. Patias
Aristotle University of Thessaloniki, School of Rural and Surveying Engineering, Thessaloniki, Greece
Email: patias@auth.gr

Ch. Georgiadis
Aristotle University of Thessaloniki, School of Civil Engineering, Thessaloniki, Greece
Email: harrisg@auth.gr

A. Kounoudes, K. Michail
SignalGenerix Ltd, Limassol, Cyprus
Email: tasos,k.michail@signalgenerix.com

***Abstract*—In everyday life, in advance to an excavation project, there is an actual prerequisite and a by regulation in many countries, to pinpoint all the underground utilities located in the project's area. In addition, it is so crucial for the clients, the public and private utilities, to have a real representation of the water, electricity, gas, sewerage, etc. underground networks.**

As the practice shows, there is a constant and growing demand for the accurate positioning of the underground utilities in order to facilitate both the utilities' and the construction companies avoid damages to the existing infrastructure. Such a damage may cause a loss of thousands of euros. It is true that many problems may arise when the digging machine operator starts digging the ground, but the most dangerous is the gas line explosions, fires or even electrocution in electric lines, water lines bursts, etc. This can be very dangerous for all the field workers in the project's area but also for the property owners; even for the people just passing nearby the area. At the same time, such a development is extremely costly either for the public or the private company or for both.

For years now there is a tangible market request for solutions that are able to effectively handle underground utilities' geospatial data and support the missions of utilities and construction companies in the real working environment. Innovative and edge technologies such as Global Navigation Satellite Systems (GNSS), sensors, Geographic Information Systems (GIS) and geodatabases, Augmented and Virtual Reality (AR/VR) can be used for monitoring, documenting and managing the utility-based geospatial and other data. In addition to this, these technologies are able to offer an intuitive 3D augmented visualization and navigation/positioning environment that can support anyone interested in the field.

LARA project, a H2020 co-funded project by the European Commission, eavesdrops these market needs and now is developing a software and hardware-based system called LARA. Understanding the real market necessities of the utilities and construction players, a group of partners, from academia and research, SMEs and end users, is implementing this multidisciplinary project. Within LARA project, the consortium partners are evolving a mobile (tablet) device to support the utility management professionals and workers on the field and under real conditions. The LARA hand-held and mobile device is incorporating state-of-the-art technologies in the domain of positioning and sensors (GNSS), AR and 3D GIS geo-databases.

The vision of LARA system is to guide the utility field workers in their daily business; to help them 'see' beneath the ground by rendering the complexity of the 3D models of the underground grid such as water, gas, sewerage and electricity. The notion is to integrate the novel components of existing technologies so as to develop an integrated navigation/positioning and information system which coordinates GNSS, AR, 3D GIS and geodatabases on a mobile platform for monitoring, documenting and managing utility infrastructures on-site. In practice, the first prototype is ready and the forthcoming period it will be tested in the real environment.

The ability and profits from the LARA system use, will be tested in two case studies scheduled during project lifetime. The two pilot applications are scheduled in Kozani (Greece) and the United Kingdom (Birmingham) with various underground utilities. The methodology for system testing has already been defined while the first results from the pilot application (Kozani, Greece) are coming the forthcoming period.

978-1-4673-8993-8/16/$31.00 ©2016 IEEE

I. INTRODUCTION

GIS is a powerful tool for handling and visualising any type of geospatial data. A current trend in Geographic Information Sciences is the development of integrated geospatial data infrastructures, which bond geographically and semantically separated geodatabases. For instance, the INSPIRE initiative [1] is the EU directive aiming to ensure that the spatial data infrastructures of the Member States are compatible and usable in a Community and transboundary context. The EU-funded pilot project HUMBOLDT [2] is a real test implementation of the standards defined in this directive. Another related EU project, is ORCHESTRA [3], which provides a framework for field data management, services for sensor access, visualisation, user-authentication and management, format conversion, coordinate transformation etc. The MELODIES project [4], originating from 'Maximising the Exploitation of Linked Open Data In Enterprise and Science' is developing eight innovative and sustainable services, based upon Open Data, for users in research, government, industry and the general public in a broad range of societal and environmental benefit areas.

The modelling of geospatial data in 3D is an active research area in geoinformation science. CityGML [5] is a specialisation of the Geography Markup Language (GML), the XML grammar for expressing geographical features. The focus of existing research is mainly targeted on above surface city models. VISTA project [6] is a prominent exception. This project aims to create integrated 3D underground models in the UK and use automated recognition for some of the modelling tasks.

Mobile GIS is the technology that gives the users the opportunity to work out of the office. Practically, it is an extension of the GIS from the office to the field by integrating several technologies: mobile and tablet devices, wireless communication and positioning systems. Mobile GIS enables on-site capturing, storing, manipulating, analysing and representation of geospatial data. In addition, mobile AR extends traditional 2D or 3D visualisations by overlaying registered visualisations over video footages. There is a clear trend towards mobile devices.

In addition, mobile AR systems are using high-precision Real Time Kinematic (RTK) GNSS. Usually, these systems use bulky hardware and limited sensor fusion. The next generation of the mobile AR prototype, the integration of a RTK GNSS receiver and performing sensor fusion with Inertial Measurement Units (IMUs) is apparent for providing a high-qualitative and functional AR experience. This is what LARA aspires to cover [7].

Several publications have been completed the last few years that are coupling existing underground utilities data with AR techniques and methods for providing x-ray observation of the underground utilities either for collision avoidance or monitoring. Much of this work has been generated by applications and devices developed in the framework of experimental research projects that try to tackle specific user needs and improve accuracy, usability, and reliability of technology. This paper attempts to describe innovative methods of integrating cutting edge technologies within the realms of AR, geodatabases, GNSS systems and positioning sensors into a mobile system. A review on the state-of-the-art of the different technologies employed is attempted and the LARA project integration framework is presented providing useful insights on methods, interoperability requirements and on site validation test beds.

II. EXISTING TECHNOLOGIES RELATED TO *LARA* PROJECT

A. 3D Geographic Information System (GIS) & Geodatabase

A GIS lets the user visualize, question, analyze, and interpret data to understand relationships, patterns, and trends. In particular, it lets the user query or analyze a database and receive the results in the form of a map. Since many data types have important geographic aspects, a GIS can have many uses: weather forecasting, sales analysis, population forecasting, and land use planning, to name a few. In addition, GIS can benefit organizations of all sizes and in almost every industry. There is a growing interest in and awareness of the economic and strategic value of GIS.

This technology can be used for managing geographic information. In particular, GIS is becoming essential to understanding what is happening and what will happen in geographic space. Furthermore, it is widely used to optimize maintenance schedules and daily fleet movements. Typical implementations can result in a savings of 10 to 30 percent in operational expenses through reduction in fuel use and staff time, improved customer service, and more efficient scheduling. GIS can also be used in order to support making better decisions about location. Common examples include real estate site selection, route/corridor selection, evacuation planning, conservation, natural resource extraction, etc. Making correct decisions about location is critical parameter to the success of an organization.

One more domain that the GIS can provide valuable help is to improve communications. Particularly, GIS-based maps and visualizations greatly assist in understanding situations and in storytelling too. They are a type of language that improves communication between different teams, departments, disciplines, professional fields, organizations, and the wider public which probably is the biggest audience.

GIS software contains a broad range of applications which involve the use of a combination of digital maps and georeferenced data. GIS software can be sorted into different categories. In the following list some of the most well-known open source or commercial applications are presented briefly [8]:

1) Open source GIS software:
    a) GRASS GIS
    b) gvSIG
    c) QGIS
    d) uDig GIS
    e) OpenJUMP
    f) MapWindow GIS
    g) SAGA GIS

2) Commercial or proprietary GIS software:
    a) ArcGIS
    b) AutoCAD/Map
    c) Geomedia
    d) Manifold
    e) Mapinfo
    f) TatukGIS

A geodatabase is highly related to a GIS system. Most of the GIS software solutions support spatial databases. A spatial database is a database that is enhanced to store and access spatial data or data that defines a geometric space. These data are often associated with geographic locations and features, or constructed features like cities, roads or pipes. Data on spatial databases are stored as coordinates, points, lines, polygons and topology. Some spatial databases handle more complex data like 3D objects, topological coverage and linear networks.

All the database systems use indexes to quickly look up values and the way that most databases index data is not optimal for spatial queries. Instead, spatial databases use a spatial index to speed up database operations. Spatial databases can perform a wide variety of spatial operations. The following list presents some of the most important operations which are specified by the Open Geospatial Consortium standard (OGC):

- Spatial measurements: computes line length, polygon area, the distance between geometries, etc.
- Spatial functions: modify existing features to create new ones, for example by providing a buffer around them, intersecting features, etc.
- Spatial predicates: allows true/false queries about spatial relationships between geometries.
- Geometry constructors: creates new geometries, usually by specifying the vertices (points or nodes) which define the shape.
- Observer functions: queries which return specific information about a feature such as the location of the center of a circle.

B. Augmented reality (AR), Virtual Reality (VR), Real-time pose estimation algorithms

Both VR and AR refer to the use of computer-generated pictures (computer graphics) for representing data-generally a rendered 3D scene. In the case of VR, the complete view is synthetic, and there is no direct link to the real environment of the user. For example, the recent advent of VR wearable devices (e.g. Oculus Rift, HTC Vive, Google Cardboard) makes it possible to place the user in a complete virtual world, where all his/her surrounding is replaced by virtual objects. By contrast, in AR, the link with the real environment is not broken. Instead, the user can visualise at the same time the real world and virtual objects unambiguously linked to the real world [9]. The real world is seen either directly in case of see-through AR (using semi-transparent AR glasses), or indirectly by using real-time video acquisition and display. The link between virtual objects and real world is in most cases the proper and seamless positioning of the virtual objects in 3D in the real scene, which implies that the exact position and orientation of the user's eye or of the used video camera - has to be computed in real-time to generate AR content. This led to the first definition of AR as ”a system that (1) combines virtual and real, (2) is interactive in real-time and (3) is registered in 3D” [10]. Thus, an AR system generally makes use of two different techniques: 3D rendering as for VR on one side, and real-time pose estimation on the other side.

Use of rendering technologies, such as programmable pipelines and shaders, allow the creation of digital physically-based materials that closely match real life counterparts in the way light interacts and reflects from the virtual objects. Due to the increased performance and parallelism of modern day GPUs, these digital materials can be applied and evaluated in real time. Real world light conditions can also be probed and used as input to such digital materials as to virtually reproduce the real light conditions [11]. This are essential to make the AR virtual objects augment the real world without looking out of place, this way breaking the immersion.

Pose estimation refers to the computation of the position and orientation of an object in 3D. This pose consists of a 3D translation and a 3D rotation and can be defined using 6 parameters. In AR, image-based pose estimation has mostly been used in the past, because AR scenarios usually require a camera as a sensor, and because minimising reprojection errors in the image leads to an optimal pose for the task of image augmentation. This type of pose estimation is referred to as inside-out tracking. Pose estimation based on image content has been achieved successfully in the past using specific patterns placed in the scene (referred to as markers) [12] or using pre-learnt natural features in the scene [13]. Recent methods such as simultaneous position and mapping (SPAM) (e.g. PTAM [14]) attempt to compute the camera pose without knowledge of the underlying scene, but for AR they require an initialization with a known reference. Recently, additional sensors have been used for computing the pose more robustly. For example, visual-inertial tracking refers to the use of an IMU coupled to the camera in a Bayesian filtering framework to compute a robust pose under non optimal image conditions[15].

C. Galileo - GPS - GLONASS - IMU sensors

The purpose of the Global navigation satellite system (GNSS) is to estimate the geographic location and velocity of an object in three dimensions. In order to estimate the location, the GNSS uses a group of satellites orbiting around the earth combined with ground-based stations. It is obvious that GNSS has unlimited applications to commercial and military industry. Two main systems are fully operational, the Global Positioning System (GPS) by USA and the GLONASS by Russia. Two other GNSS systems are under development, the Galileo is Europe's GNSS and the BeiDou by China ([16]). The LARA system is compatible with all commercial GNSS systems. The estimated geographic location accuracy is affected by a number of errors including ionosphere, multipath etc. (such error can go up to 10m, see [17]) therefore methods have been developed to eliminate such errors hence improve accuracy. The accuracy can be reduced at the level of one meter using satellite-based augmentation system (SBAS) down to a few centimetres using differential positioning systems ([18]). High precision receivers can achieve a centimeter using state-of-the-art methods like RTK ([19]) and Precise Point Positioning (PPP) ([20], [21]). The LARA receiver is equipped with a high precision, low power with long autonomy GNSS receiver module able to achieve accuracies at the level of centimeter. The receiver is able to work with multiple constellations including the Galileo and EGNOS systems.

The orientation of the LARA receiver is also essential therefore it is also equipped with an IMU module able to estimate the orientation in three dimensions (Roll-Pitch-Yaw). Several orientation sensors exist in the market for such estimation. The LARA device is equipped with a state-of-the-art high precision orientation module.

III. THE *LARA* PROJECT APPROACH

LARA is a H2020 co-funded project by the European Commission which aspires to develop a system, both software and hardware-based, by integrating different innovative components of existing technologies. The aim of LARA is to design and develop an integrated navigation/positioning and information system which coordinates GNSS, AR, 3D GIS, and geodatabases on a mobile platform for monitoring, documenting and managing utility infrastructures on-site. The LARA system will be able to guide utility field workers to locate the working area by helping them see beneath the ground, rendering the complexity of the 3D models of the underground grid such as water, gas and electricity.

The project sets four (4) concrete and measurable objectives in developing a highly innovative and adaptive application taking advantage of the Galileo and EGNOS.

1) Design, develop and exploit an entirely practical mobile device for utility field workers
2) Make use and integrate mobile AR interfaces
3) Improve on site interaction and navigation application
4) Commercialise the final product and the related services as well as address relevant markets

A. 3D GIS and Geodatabase

The 3D GIS module is a costume GIS application which is designed and implemented in order to cover the needs of the LARA platform and offers a user - friendly and easy to use software. In practise, the 3D GIS development is made in Java by using an open source well-known library, namely GeoTools [22]. The consortium decided to use Java as a programming language for the GIS module due to the characteristics of this specific language, such as:

- Java is object-oriented
- Java is platform-independent
- Java is robust
- Java is multithreaded

Furthermore, GeoTools is a free software (LGPL) GIS toolkit for developing standard compliant solutions. It provides an implementation of OGC specifications as they are developed. It is written in Java and currently is under active development. It is used by GeoServer, uDig, Geopublisher, 52N, Geomajas and many proprietary projects [23]. The core features which are supported by GeoTools library are listed below:

- Interfaces for spatial concepts and data structures
- JTS Topology Suite Geometry
- Attribute, spatial and temporal filters matching the OGC Filter specification
- Decoding technology (with bindings for manyGML, Filter, KML, SLD, and SE and other OGC standards).

The GeoTools library forms a stack for the core modules; with additional plugins available to support additional formats, extensions and integration options [24].

Figure 1 shows the physical view of the 3D GIS module as designed and implemented for the *LARA* system.

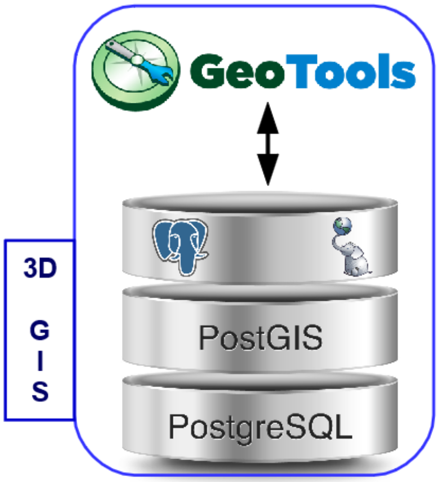

Fig. 1. Physical view of the 3D GIS module.

The main services which are supported by the 3D GIS module are listed below:

- GIS viewer
- Load shapefiles / Add / Edit multiple layers
- Support different coordinate reference systems (CRS)
- Handle shapefiles
- Support a base map - add WMS layer to the GIS viewer
- Add raster layer
- Show user's position
- Connect to the geodatabase / Store shapefiles
- Edit / Delete information from the db
- Support a back office update system

The geodatabase is supported by the PostgreSQL and PostGIS. PostgreSQL [25], often simply Postgres, is an object-relational database management system (ORDBMS) with an emphasis on extensibility and standards-compliance. A very important characteristic of the PostgreSQL is that is a cross-platform and runs on many operating systems (OS) including Linux, FreeBSD, OS X, Solaris, and Microsoft Windows. The PostGIS [26] is an open source software (OSS) program that supports geographic objects in the PostgreSQL object-relational database. PostGIS follows the simple features model for SQL specification from the OGC. As a result, the combination of the PostgreSQL and PostGIS is an excellent solution to support spatial data. Figure 2 presents the main architecture decisions regarding the geodatabase module.

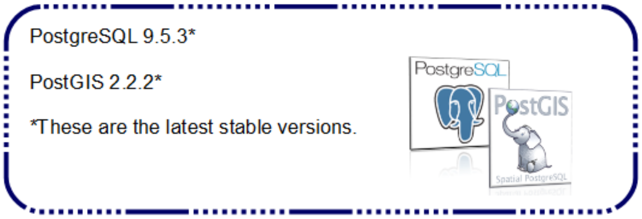

Fig. 2. Geodatabase technologies.

B. AR & VR

The AR and VR module is responsible for the presentation of the result as an augmented view of the real environment. The input of this module consists of pieces of information provided by other modules of the system:

1) The 3D GIS module provides a description of the local utilities network in form of a shape file (in which the pipes are represented as segments between geolocalised 3D points)
2) The Sensor Fusion (SF) module provides the pose of the camera as computed from the GNSS signal and other sensors (IMU and control points in the camera field of view)
3) The current image is provided by the rear camera of the tablet. The AR and VR module processes this information in a number of steps as shown in Figure 3.

First, the shape to collada converter transforms the shape information into a realistic 3D model representing network utilities as 3D pipes with a specific radius and a digital material that imitates the real life materials (plastic, metal, etc). This conversion also called tessellation - is crucial for the AR effect, as the network utilities have to be visualized as 3D virtual objects (see Figure 4). Second, when using a renderer, a virtual camera has to be created to render the scene. This virtual camera can be positioned arbitrarily in the scene, and its orientation (e.g. pitch, roll, yaw) can be chosen at will. In addition, the virtual camera can simulate various fields of view and zooms. Thus, it is clear that the virtual camera used in a renderer has basically the same parameters as a real camera has. The whole concept of AR is based on the idea of creating a virtual camera with the exact same parameters as the real camera that generated the background image. If this is the case, the 3D objects will appear seamlessly integrated into the 3D scene (at least from a geometric point of view accurate simulation of light conditions is out of the scope of the project).

Therefore, the pose and camera calibration information of the Positioning Receiver / Sensor Fusion (PR-SF) module has to be converted into the equivalent matrices required by the rendering engine. In this process, the provided rotation angles and camera position are converted into an OpenGL modelview matrix, and the camera interior orientation parameters are converted into an OpenGL projection matrix. These two matrices can be fed as input to the rendering engine for positioning the virtual camera. Third, the AR scene has to be composed before rendering. In this process, a scenegraph containing two different cameras is created: the background image is rendered as a texture on a plane as seen by a virtual camera with orthographic projection. The projection is chosen such that the plane fits exactly the dimensions of the output image. The 3D elements (virtual objects representing the network utilities) are rendered as 3D objects seen by a second virtual camera with perspective projection. The position and orientation of this second camera is defined by the OpenGL modelview matrix above, and the projection parameters are the ones derived from the interior orientation of the real camera. The result of rendering this scenegraph is that the real image appears as a background and the 3D objects are rendered as seamlessly integrated into this background.

Figure 5 shows an example of scene composition. In the rendering process, the pipeline has been converted from an on-screen rendering paradigm to an off-screen rendering paradigm. This means that the rendering of the virtual objects as well as the image composition with the real video frame as background is made entirely in memory rather than on the screen as usually done in rendering applications. This proves to be extremely useful for embedding the AR module into a

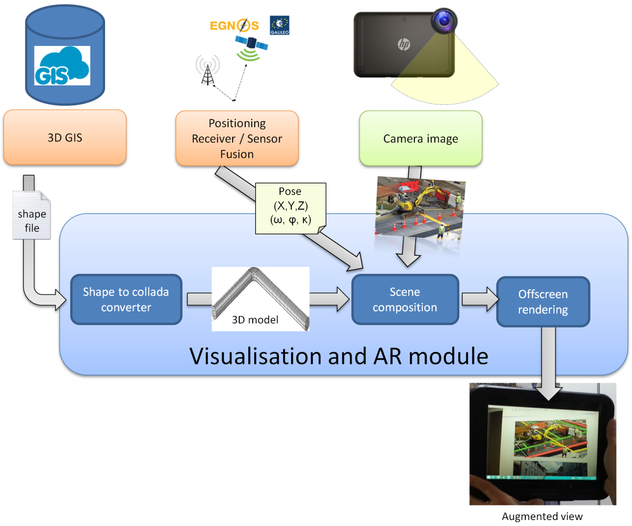

Fig. 3. Components and dependencies of the AR & VR module

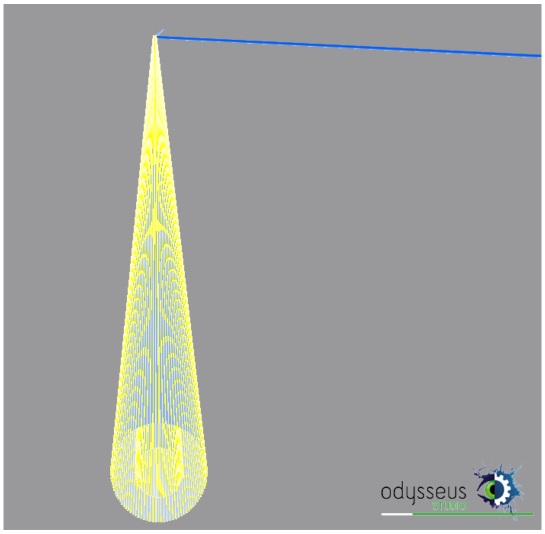

Fig. 4. Example of tessellated pipe (only virtual part)

larger application that manages the Graphical User Interface (GUI) separately.

Fig. 5. Example of of scene composition with real background image

C. Galileo - GPS - GLONASS - IMU sensors

A simplified block diagram of the geolocation receiver is illustrated in Figure 6. It consists of a GNSS module, an IMU module and the GUI. The LARA device is able to receive multiple constellations (i.e. GPS, GLONASS, BeiDou) including the European state-of-the-art GNSS system called Galileo which is designed to improve the accuracy, increase satellite availability and integrity. The satellite availability and integrity is enhanced due to the large number of satellites but further improvement of the accuracy is a necessity therefore differential corrections technology (EGNOS, DGNSS and RTK) are leveraged in order to reduce the geolocation error to around 1cm. The receiver is equipped with an IMU module that is used for the estimation of the device orientation. The high precision geolocation estimation is based on sensor fusion algorithm. The fusion is done using measurements from tri-axis accelerometer, magnetometer and gyroscope. Flexibility and long autonomy of the LARA device is targeted in order to promote the LARA device in the market able to communicate through high level user interface compatible with most commercialized tablets. The prototype of the receiver has been developed and is shown in Figure 7, 8.

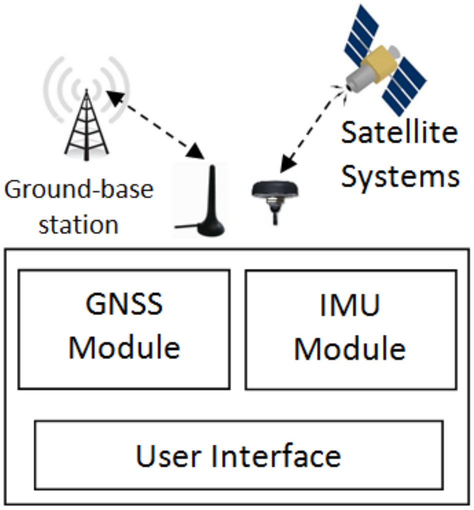

Fig. 6. LARA geolocation receiver

Fig. 7. Front view

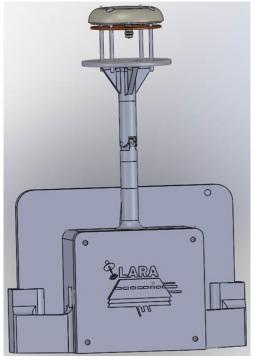

Fig. 8. Back view

D. Real-time pose estimation algorithms

In the LARA approach pose estimation parameters are initially recovered through the use of GNSS and IMU sensors. GNSS provides the location parameters, while IMU provides the attitude parameters. In the case that the recovered parameters' accuracy is lower than the system specifications, a photogrammetric pose estimation module is triggered. The photogrammetric approach (photogrammetric resection) [27], [28] uses at least 3 control points to compute the pose estimation parameters. In cases when there is lack of control points (less than 3, or only one) pose estimation parameters can be computed using vanishing points, a control point and a vertical or horizontal distance originating from the control point.

E. The integrated LARA solution

1) Working methodology: The project has followed a user-centred design (UCD) methodology to ensure that end user needs are fulfilled. Studies with stakeholders during the first months of the project (before the development phase) led the consortium to obtain user and technical requirements. These have served as set of constraints that helps to identify the aspects that the prototype development should focus more on in order to satisfy real end user's needs.

The LARA system has been developed through a (technology centred design) continuous development process in order to comply with the real industry needs. Three prototypes were planned during the 24 months of the project, each one corresponding to a set of functionalities during the continuous development process. This will allow an active collaboration of the users on the final product, ensuring that the final prototype will mature enough for the stakeholders to use it on a real environment.

The complexity of the system has been gradually increasing with each version. Prototype V0.5 served as an early demonstration and proof of concept, while the first prototype (V1.0) includes a basic set of functionalities, allowing to receive feedback in order to, evaluate and validate the prototype through lab trials and pilot tests, at an early stage of development. Following UCD methodology implies that pilots are our key tests to evaluate the system. Thus, the final prototype (V2.0) will not only implement every functionality; it will also include an enhanced version of each module based on our experience with the end users during the pilots.

2) Internal architecture: LARA system requires its modules to constantly exchange information, this mutual dependency between modules along with different programming languages (mainly C++ and Java) and input data format, implies that there must be a layer orchestrating the process. Since the UI module has to establish contact with every module to display their data, the consortium decided that it was the UI module the most appropriate for the orchestration of the system.

LARA system is composed by the following modules: Positioning receiver (PR), Sensor fusion (SF), User Interface (UI), 3D GIS Geodatabase (3DG) and Visualisation and AR (AR).

3) Interfacing and data flows: To ensure the information flows between modules as expected, and due to the different nature of every module, each one must define its own interface, i.e., the data it needs as input and what will it return as output, as well as the required format for the data. Figure 9 below illustrates the interfaces defined as well as data flows.

The PR and SF modules work jointly since both are implemented on a microcontroller built on an external board, in order to have fluid and fast access to the raw data from the sensors. The PR gathers the raw data from the sensors which the SF then processes, and finally, the data is stored into the database. This process must be at a high enough rate, so that delays wont affect system performance. The UI retrieves the processed data as well as some flags about the accuracy of the measurements through the PR-SF-UI interface, and sends it to the different modules that use it as input.

The 3DG uses shape files as input, which contain information about the underground infrastructure network, not just about the location of the pipes but also about their attributes (e.g. material, diameter, length, etc.). This module uses the current location to centre the screen on the user's position as well as to generate a clipped version of the shape files around the user's location.

After this process, the AR module can generate a 3D model based on them. Correlating the 3D model, user location, device pose and camera calibration parameters, the module is able to render a view of the pipes over a raw image from the camera.

The UI is responsible for acquiring the information from these modules and displaying it to interface with the end users on a simple, attractive and intuitive way such that an average tablet user would be able to understand it. This has been achieved by following the mentioned UCD methodology.

The orchestration layer inside the UI is responsible for starting each module when it is required, as well as obtaining the output from each module and applying the conversions needed to adapt it for the specific format. The user's input is captured by a controller implemented in Java, interacting with the rest of the modules' interfaces to act as an orchestrator.

4) Physical architecture: Since the LARA solution requires the input from several sensors in order to achieve a highly accurate positioning system, not present on standard tablets, an external printed circuit board (PCB) integration the required hardware has been developed. This custom hardware module has both USB and Bluetooth v4.0 interfaces, so it could be connected virtually to any machine, ensuring again the portability of the LARA system to a different tablet or OS. Therefore, this sensor board will be used along with the rest of the hardware that is included, to assure an error below 20 cm on position and 1 degree on the rotation angles. Figure 10 illustrates the physical architecture of the LARA solution.

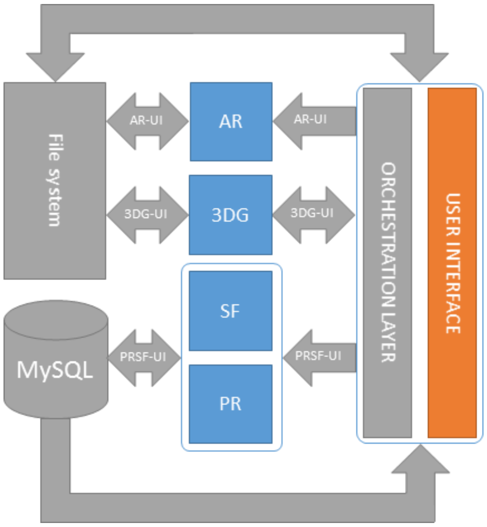

Fig. 9. LARA data flow

Every single module of the LARA system is fundamental on its own, to achieve the ultimate goal: augmenting the underground infrastructure over raw images from the camera, to assist the end users on the work field. However, none of these modules would be able to perform their whole set of functionalities without an orchestrator. Hence, the orchestration layer is as important as any other module, since it acts as a kind of glue that allows the different subsystems to work together, ensuring the information flows as expected, thus allowing the correct operation of each module individually and the system as a whole.

5) Implementation: LARA solution is divided into several modules with different programming languages. The UI module (which contains the orchestration layer) has been implemented in Java due to it can be easily integrated with most languages. Java code can also run on any machine or OS, provided that a JVM has been implemented for that platform. This software portability, along with a portable sensor board with a common USB or Bluetooth v4.0 interface, opens up a lot of possibilities for expanding the LARA solution to different devices.

Thus, the UI has been implemented using JavaFX [29], a library that is included in Java8 which implements the MVC pattern, creating a central controller that captures user interaction and establish the starting point of the orchestration process. The 3DG module is also implemented in Java, and it is based on Geotools and PostGIS to provide geospatial objects on a PostgreSQL database. Geotools implements its views as Swing components, which can be easily integrated inside JavaFX.

The PR-SF subsystem runs on a microcontroller and accesses the data from the sensors. To communicate with this external hardware, a background service will run on the tablet, listening for new commands inserted on a MySQL database table. These commands will be inserted by the UI module through stored procedures. Once the positioning module receives a new command, it performs the requested operation and stores the result into the database, from where the UI can retrieve it and send it to the module that needs it.

The AR module is also written in C++ (due to the high performance it requires) and compiled as console applications, the communication with this module uses text files. This subsystem is in fact split in two submodules: the first one generates a 3D model from a shape file; the second one uses this model, a raw image from the camera and the pose of the user, and renders a 3D model of the pipes over the image.

Fig. 10. Lara architecture

6) System orchestration: In order to provide all the functionalities of the system, all the modules must work together. This implies several information flows that need to be orchestrated by the UI module in order to make the LARA solution work integrated as a whole. The best example is the AR engine. In order to display the underground infrastructure on AR, the renderer module needs the following information as input:

- Raw image from the camera
- 3D model of the pipes (generated from a clipped shape file via one of the AR submodules)
- Camera parameters (extracted from the calibration during the initial setup of the system)
- Pose estimation (extracted from the database as inserted by the PR-SF background service)

Figure 11 illustrates the workflow that must be followed in order to augment a static image.

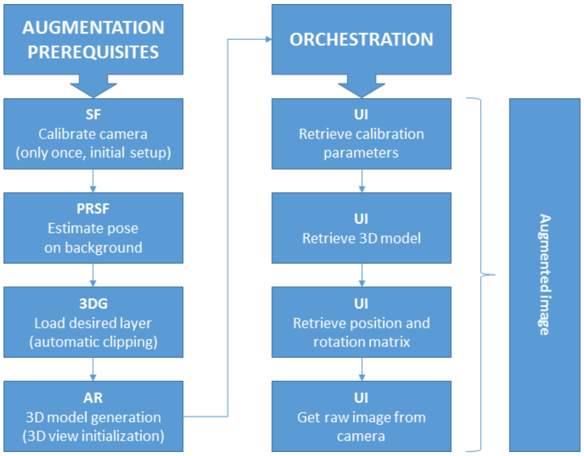

Fig. 11. Augmenting image workflow

7) LARA solution: As a result, the LARA system provide to the site personnel with a meaningful tool that will enable them on the work field. First, by locating the work zone on a GIS viewer, they can obtain a visual representation of the underground infrastructure and its attributes. Then, stakeholders can take advantage of the augmented reality engine to have an intuitive view of this infrastructure on 3D over a real image from the camera. Finally, after the work is done, they can use this same tool to generate reports to update the existing maps, thus updating the actual infrastructure information after a work is performed

IV. SUMMARY AND FORTHCOMING WORK

LARA project is developing a software and hardware-based system, under a H2020 project which is co-funded by the European Commission. The system is capable of effectively manage underground utilities' geospatial data. LARA is combining different submodules in order to develop an integrated navigation/positioning and information system. This system is coordinating different technologies, such as GNSS, AR, 3D GIS and geodatabase on a mobile platform (tablet) for monitoring, documenting and managing utility infrastructures onsite. In practice, the LARA system is considered to support the utilities' field workers in locating the working area by helping them see beneath the ground and rendering the complexity of the 3D models of the underground grid such as water, gas and electricity.

At this stage, LARA project is ready to present the V1.0 prototype and this period is running the first pilot in one of the pilots' cities in Kozani, Greece. The first results are expected very soon. We scheduled to run one more pilot, a couple of months before the official project end, in Birmingham city. Next, the final V2.0 prototype at the end of the project will be ready.

The effectiveness and the capabilities of the LARA platform are planned to be verified in two real case studies located in Greece and the United Kingdom. Both case studies will take place with various underground utilities. The results from these pilots will feed us with useful tips to be used for improving the performance of LARA platform.

The final stage of the project concerns the definition of the business models to be used for LARA system commercialisation.

ACKNOWLEDGMENT

The LARA project leading to this application has received funding from the European GNSS Agency under the European Union's Horizon 2020 research and innovation programme under grant agreement No 641460. LARA is co-funded for the period 2015-2017.

REFERENCES

[1] INSPIRE, http://inspire.jrc.ec.europa.eu, Last accessed 17 September 2016

[2] HUMBOLDT, http://www.esdi-humboldt.eu, Last accessed 17 September 2016

[3] ORCHESTRA, http://www.eu-orchestra.org, Last accessed 17 September 2016

[4] MELODIES, http://melodiesproject.eu, Last accessed 17 September 2016

[5] T. Kolbe and G. Groeger and L. Pluemer, *CityGML Interoperable Access to 3D City Models*,Proceedings of the Int. Symposium on Geoinformation for Disaster Management, Springer, 2005

[6] A. R. Beck and G. Fu and A. G. Cohn and B. Bennett and J. G. Stell, *A framework for utility data integration in the UK*, Coors, V., Rumor, M., Fendel, E. M. & Zlatanova, S. (Eds) Urban and Regional Data Management - Proceedings of the Urban Data Management Society Symposium, 2007

[7] E. Stylianidis and E. Valari and K. Smagas and A. Pagani and J. Henriques and A. Garca and E. Jimeno and I. Carrillo and P. Patias and C. Georgiadis and A. Kounoudes and K. Michail, *LBS AUGMENTED REALITY ASSISTIVE SYSTEM FOR UTILITIES INFRASTRUCTURE MANAGEMENT THROUGH GALILEO AND EGNOS*, ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, pp. 1179-1185, 2016

[8] S. Steiniger and E.Bocher, *An Overview on Current Free and Open Source Desktop GIS Developments*, Retrieved 2011-08-05

[9] A. Pagani, *Modeling Reality for Camera Registration in Augmented Reality Applications*, Knstliche Intelligenz (KI), 2014

[10] Azuma, R.T. and others, *A survey of augmented reality*, Presence: Teleoperators & Virtual Environments, pp. 355–385, 1997

[11] P. Debevec and P. Graham and J. Busch and M. Bolas, *A Single-shot Light Probe*, ACM SIGGRAPH 2012 Talks, SIGGRAPH '12, pp.10:1–10:1, 2012

[12] A. Pagani and J. Koehler and D. Stricker, *Circular Markers for Camera Pose Estimation*, Proceedings of the International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS), 2011

[13] M. Zoellner and A. Pagani and Y. Pastarmov and H. Wuest and D. Stricker, *Reality Filtering: A Visual Time Machine in Augmented Reality*, Proceedings of the International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST), 2008

[14] G. Klein and D. Murray, *Parallel Tracking and Mapping for Small AR Workspaces*, Proceedings of the IEEE International Symposium on Mixed and Augmented Reality (ISMAR), 2007

[15] G. Bleser and D.Stricker, *Advanced tracking through efficient image processing and visual-inertial sensor fusion*, Computer & Graphics, 2008

[16] A. Madry and Scott, *Global Navigation Satellite Systems and Their Applications*, Springer, pp. 1-8, 2015

[17] M. Petovello, *How do measurement errors propagate into GNSS position estimates?*, InsideGNSS, pp. 30-34, 2014

[18] W. Gerhard and B. Andreas and S. Gimter and B. Volker and P. Hankemeier, *Reducing distance dependent errors for real-time precise DGPS applications by establishing reference station networks*, Conference proceedings of ion GPS,Institute of Navigation, pp. 845-1852, 1996

[19] L.Kislig, *What is a virtual reference station and how does it work?*, InsideGNSS, 2011

[20] O. Chassagne, *One-Centimeter Accuracy with PPP*, InsideGNSS, pp. 49-54, 2012

[21] G. Lachapelle and M. Petovello and Y. Gao and L. J. Garin, *Precise Point Positioning and Its Challenges,Aided-GNSS and Signal Tracking*, InsideGNSS, pp. 16-12, 2006

[22] GeoTools, http://www.geotools.org/, Last accessed 17 September 2016

[23] GeoTools WIKI, https://en.wikipedia.org/wiki/GeoTools, Last accessed 17 September 2016

[24] J. Garnett, GeoToolsUserGuide, Open Source Geospatial Foundation, Retrieved 3 October 2011

[25] PostgreSQL, https://en.wikipedia.org/wiki/PostgreSQL, Last accessed 17 September 2016

[26] PostGIS, https://en.wikipedia.org/wiki/PostGIS, Last accessed 17 September 2016

[27] A. Dermanis, *Analytical Photogrammetry (in greek)*, Ziti Editions, 1991

[28] P. Patias, *Introduction to Photogrammetry (in greek)*, Ziti Editions, 1993

[29] JavaFX Documentation, Oracle, http://docs.oracle.com/javase/8/javafx/get-started-tutorial/jfx-overview.htm#JFXST784, Last accessed 17 September 2016