Chapter 22
Graph-Based Model for Context-Aware Maintenance Assistance with Augmented Reality and 3D Visualization

Michael Abramovici, Mario Wolf and Matthias Neges

Abstract The benefits of augmented reality applications in maintenance are widely known and since high-performance smart devices are the common standard for mobile devices, the actual preconditions for the usage of such applications seem promising. Problems emerge whenever service is to be conducted in an area of banned photography. Using a smart device with a camera is either simply not allowed, or the internal camera has to be pasted over to allow entrance into the restricted areas. Either way, the technician still relies on his maintenance assistant system to provide useful information if he does not want to go back to pen and paper. In this article a concept is elaborated that offers context-sensitive guidance, a highly dynamic data model and different views, depending on the availability of an internal camera and/or restrictions of the work environment. The approach presented was implemented and validated under laboratory conditions with a complex hydraulic system as a demonstrator machine. The prototype will be the foundation of an industrial case study concerning the combination of IoT enabled machinery and smart devices in maintenance later this year.

22.1 Introduction

Augmented and virtual reality are current trends in the development of maintenance assistant systems (cf. [1, 2]), while 3D computer graphics are state of the art in product development and visualization. As the benefits in presentation and interaction are widely known these technologies are used for innovative solutions for mobile applications [3]. Schlick et al. [4] analyzed training success based on the instructional media and found that graphical instructions are superior to textual ones whenever the repetition rate is low. As mobile devices (smart phones, tablets) have

M. Abramovici · M. Wolf (✉) · M. Neges
Ruhr-University Bochum, Universitätsstraße 150, 44780 Bochum, Germany
e-mail: mario.wolf@itm.rub.de

367

fully saturated the market [5] throughout the last decade, augmented reality applications are ready to be used on a large scale.

Augmented reality is defined [6] as overlaying reality with additional virtual content and as such relies on camera footage, which disqualifies the whole area of augmented reality applications for usage in company sites with a ban on photography. Especially ancillary industries are often bound by contract to facilitate such zones to prevent industrial espionage. The question arises whether the camera-reliant augmented reality application's content could be used for a camera-independent mode without reducing the application's benefits.

Due to networking capability of products in the Internet of Things (IoT) and the accompanying availability of data, products can not only operate, react and start actions autonomous but even use additional services and evaluate the data received from external sources [7]. This among others offers enormous potential in maintenance of technical machinery, like the improvement of service documentation by combining it with data from the actual machines [8, 9].

A context sensitive support of maintenance with Augmented Reality (AR) for instance reduces worktime by 55% in a complex pipe-systems while at the same time halves the error rate [10]. Augmented Reality applications allow an interactive problem-solving, which enable intuitive work with an interactive guide that is dependent on the actual state of the smart product, rather than static step by step instructions.

Generally, in the context of maintenance assistance, there are certain common features between augmented reality applications and applications that display 3D models to enrich the information for their recipients. Such is the need for the preparation of work instructions, the usage of 3D mockup models instead of full-fledged Computer Aided Design (CAD) models and the need for precise orientation of the afore mentioned models [11].

When taking into account the similar functionality of augmented reality and 3D model displaying applications, a situation-dependent visualization mode should be implementable.

In this paper, we

- determine requirements for a camera-independent mode in an AR application,
- define the planned functionalities,
- describe the theory behind marking technology-based identification, how it can be used for adaptive 3D-Model orientation and
- validate our approach with two use cases in each visualization mode.

The result will be presented as prototypical solution consisting of a back-end service for data acquisition, data analysis and a front-end AR application to support the maintenance technician in the field. Since the potential of the use-phase of Radio Frequency Identification (RFID) and Augmented Reality applications have been treated already [12, 13], this chapter will focus on the dynamic data-model and the multi-functional maintenance assistant.

The first use case includes the visualization of real-time operational data, while the second case includes the actual maintenance assistance with context-sensitive status analysis and indications. To realize context awareness, we use our validated approach on graph-based algorithms [13] to guide the technician after evaluating the current state of the machine, whether it is in regular AR- or camera-independent mode.

22.2   Aims and Requirements

The aim of the presented chapter is to better support technicians on-site through smart usage of existing information and mobile devices, even in zones of banned photography. To achieve that, the range of functions of the components discussed in [13] need to be extended, either to improve shortcomings or to enable new functionalities.

The main goals are to minimize the need for manual inputs, offer real-time information about the machinery at hand in a manner that is quick and easy to grasp. All that should work independently of the availability or legitimacy of a camera on the smart device.

Taking advantage of the Internet of Things (IoT, cf. [14]) the current approach will no longer contain the error-prone (cf. [15]) manual gathering of the operational data but analyzes real-time sensor data directly read from the machine at hand and evaluated trough modules within the IoT platform.

The Goal of expanding the concept is the inclusion of information from various source-systems like Product Lifecycle Management (PLM) and Computerized Management Systems (CMMS). These types of applications usually are designed to be used on a desktop PCs, with some concepts to bring them to mobile devices without enhancing the functionality [16].

Joint with the possibility of being able to analyze real-time sensor data of the momentary machine status, potential arises for dynamic responses of the assistant application in form of indications, hints and warnings displayed in augmented reality [17] or 3D during a maintenance task.

After completing the maintenance task an automated documentation and creation of work performance record is conducted to minimize most of handwritten work by the technical personnel. The following synchronization of mobile application and the graph-based model provides a way of efficient data management and feedback repatriation of data that exceeds the regular handwritten coverage in both coverage and future-proofness. This way machine or component specific data for the manufacturer or contractor can be retraced to enhance attrition calculation or product usage data for the product development of the successor generation (cf. [9, 18]).

The assistance system is dependent on sensor data, so our demonstrator had to be fitted with according devices. Particularly interesting are wireless sensors on basis of Zigbee Standards (IEEE 802.15.4) or Radio Frequency Identification (RFID) ISO 18000-6C, as these are easy to install and configure in a prototypical

environment. Theses sensors then must be equipped to send their payload to the cloud-based IoT platform for further processing.

As consequence of the desired functionality as part of the Internet of Things (cf. [19]) an active internet connection is needed for communication of the integration platform in order to access the real-time operational data and make use of the analytics functions.

As further electronic documentations and 3D-perspectives of the concerning components shall be made available, the selection of the possible viewing options is crucial.

To get a grasp on the problem of representing certain information, one must understand the specialties of the different projection modes. As the unrestricted viewing mode shall be augmented reality, the functional range of AR is the reference point. Broll [20] defines augmented reality as the enhancement of reality with artificial digital content, while the interaction is real-time capable and interactive. Furthermore, the enhancement must be continuously adapted to the user's perspective to enable the fusion between digital and real elements. Intuitively, when taking away the camera picture what is left of augmented reality is the digital content alone, which means 3D computer graphics in this context.

On the other hand, following the Milgram's reality-virtuality continuum [21] from augmented reality in the direction of *less reality*, the final destination is virtual reality (Fig. 22.1).

Broll [20] characterizes 3D computer graphics as a purely visual non-immersive presentation, that is not time-sensitive, user independent (exocentric perspective), displaying only a static scene with 2D-Interaction with i.e. Mouse and Keyboard. By contrast, virtual reality applications feature multimodal presentation (visual, acoustic, haptic), real-time interaction, user centered, immersive presentation (egocentric perspective) and 3D-Interaction (hand-, head- and body-movement).

The desired solution for the presented approach shall offer the benefits of augmented reality and virtual reality in regard of real-time capable interaction through movement and gestures, egocentric perspective and information density, while neglecting immersion for ease of use with smart devices without head mounted displays (HMD). As 3D models are available for overlay-display in augmented reality, these models could be used in a camera-less visualization mode if the maintenance is to be done in an area of prohibited camera use.

Fig. 22.1 Adapted reality-virtuality continuum based on Milgram et al. [21]

22.3 Concept

The overall approach is subdivided into the three modules *Cloud Data Sourcing*, *Cloud Data Handling* and *On-site Maintenance Assistant* to represent each functional unit and labeling the distribution pattern. Each module is then further subdivided into two layers to separate distinct functionalities within said functional units. Conceptually the flow of payload data, meaning data that is required to enable the technician on-site to get the actual information he needs, is strictly bottom-up (cf. Fig. 22.2). As the technician is not only the recipient of information, but also the author, the concept must therefore allow feedback data to flow back into the responsible components for later use.

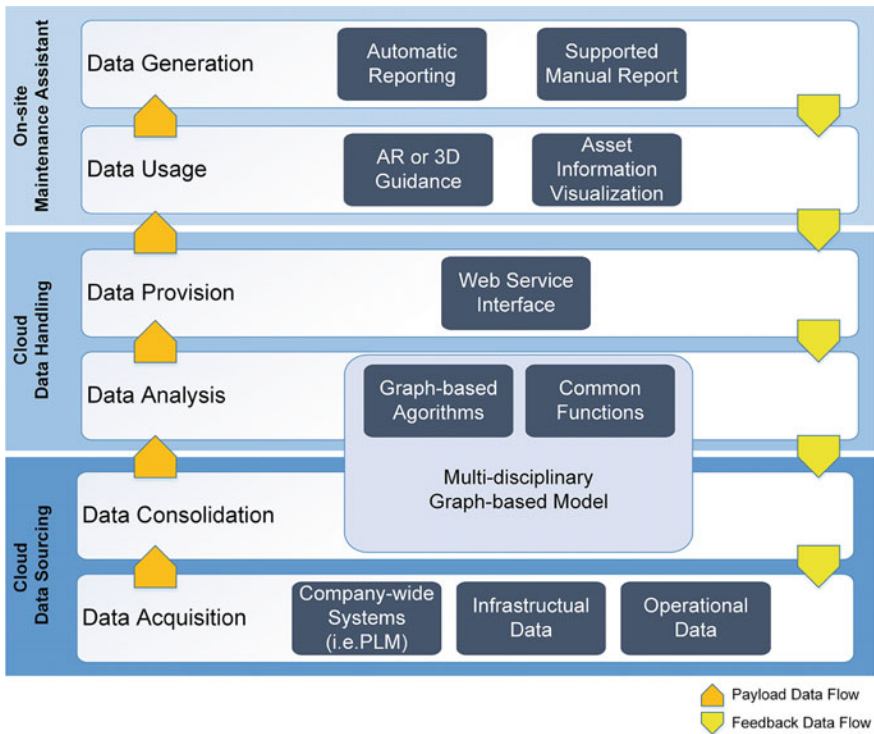The three main modules will be explained further in the following subsections.

Fig. 22.2 Conceptual system architecture

22.3.1 Data Sourcing

In this module, data sources for processing in the multi-disciplinary graph-based model can be found, as well as the mechanisms for bringing the data into an evaluable form.

The *Data Acquisition* layer gathers data particularly from PLM and CMMS as this information is needed for the technician's work on-site. Along with the data extraction from the company-wide systems the interfaces are used to backchannel feedback information from the maintenance processes into the graph-based model and from there into the company-wide systems thus ensuring that the company-wide systems possess the leading data basis and are up to date.

The operational data is not only gathered from sensors of individual machines but also from various other services like location-based services (i.e. weather forecast), ambient data from the facility, availability data for surrounding systems or similar context-relevant data. As the administered amount is high, consolidation and filtering of the data is needed, before an actual interpretation for the use through a technician.

Haberfellner and Daenzer [22] cite that models as a depiction of reality must be convincing enough in regards to the situation and problem position. This means that in all deliberations the question of functionality and problem relevance is to be asked. To answer the question of functionality and relevance Harrison [23] cites the strength of graph databases lie especially on the use cases in which the correlation of elements is more important than the elements themselves.

While company-wide systems specialize on administration of inventory data, the multi-disciplinary graph-based model is used as external storage for the maintenance data and information extracted and aggregated from existent systems, to be dynamically evaluable for service technicians. To control the complexity of the model creation (cf. [24]) domains which are involved in the setup of the model are limited to product structure, electric, pneumatic and hydraulic networks, as well as task management.

It should be clarified that the functionalities of this approach are based on the considerations of Harrison, Haberfellner and Daenzer, which lead to the following questions:

- Which *things* are relevant to the activity? Which *things* are affected by the planned activity?
- Do *connections* exist between those things?
- Are there *rules* to be identified in order to evaluate connections between things in an algorithmic fashion?

If this basic idea is transferred into the world of graph theory things are consequently displayable as vertices. These vertices represent certain entities with all their attributes. Connections between the individual things will be mapped with edges. Edges in graphs can have attributes on their own. Constraints ensure the
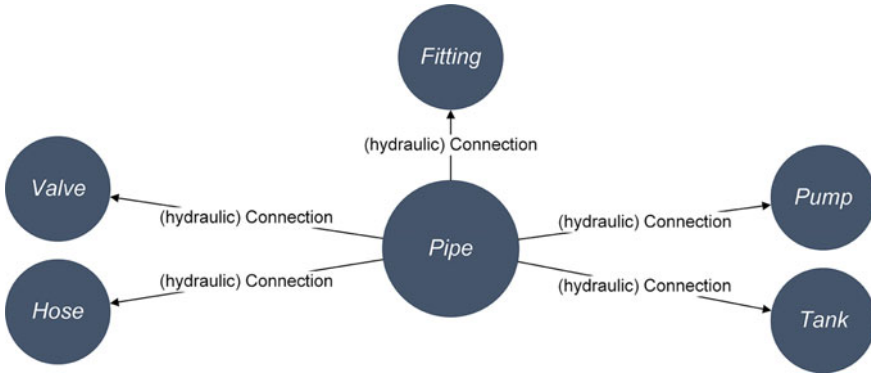
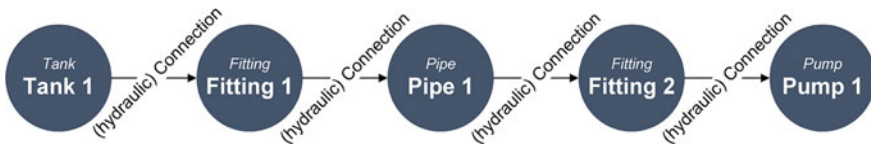**Fig. 22.3** Exemplary valid outgoing connections between hydraulic components

Fig. 22.4 Exemplary network of distinct hydraulic components

necessary rules for validity of connections, so that only certain types of vertices may have certain type of in- and outgoing edges.

An example of implementing this order is displayed in Fig. 22.3. It is to show the valid outgoing connections between various hydraulic components from the vertex of type *Pipe* and elucidate the appropriate rules by reference on a clear example.

Following the pattern vertices of type *Pipe* are solely entitled to outgoing edges of type *(hydraulic) Connection* to the types *Valve, Hose, Fitting, Pump* or *Tank* as their target.

The classification of relevant components of maintenance focuses on product characteristics rather than on product attributes like stated in [24]. Particularly the characteristics which can be identified quickly and safely on-site by the technician should be chosen, such as allocation of a certain functional groups, for instance types like seen in Fig. 22.3 and similar discrete physical object types. Types in this context generally can be understood as an equivalent of classes in object-oriented programming. This means once a class is designed, you can create discrete, distinguishable objects of the same pattern, but with different characteristics, i.e. two pipes with different lengths. An example for valid instantiation of said classes can be seen in Fig. 22.4.

The next step in the process of building a product-characteristic-based graph presentation consists of the modelling of classes and their subsumption in hierarchies. These form the following perspectives of the data:

- Hierarchy of the vertex classes
- Hierarchy of the edge classes
- Constraints between vertices and edges

To ensure a repeatable identification capability the classification of components was arranged according to the component's functional characteristics in context of maintenance. The same procedure was performed on the relations and their corresponding constraints to avoid erroneous entries. Due to these arrangements a combination of an electric component as target of a hydraulic flow is (fortunately) impossible.

These preparations build the basic frame for the graph-based model, which is by design sound in itself and allows considerations of all relevant elements in the maintenance process. Once the data is collected from the source systems, the utilization of said objects of classes of different relevant elements in maintenance can begin and thus *Data Consolidation* is provided.

22.3.2 Data Handling

Once the model is set up utilizing the rules mentioned in the last section it can be used to gain insight about the status of machines at hand. The *Data Analysis* layer denominates the entirety of the provided functions and algorithms to interpret or manipulate the data in the graph-representation. Using a mixture of classic relational query design and graph-based algorithms (cf. [13]) like the Dijkstra or Floyd-Warshall, one is able to find paths between relevant start and end point in the graph network.

For instance, an analyzation can be started by following hydraulic connections between relevant elements of a facility (i.e. a cooling cycle consisting of a multitude of parts), which translates to finding relevant vertices of correct type and following the right classes of edges to form a path.

If basis and goal are known a pathfinding of the graph is possible without additional features of edges and vertices. The addition of such information enables either more precise results or concrete conclusions based on further algorithms.

If the actual representations of functional elements in the graph are conducted as displayed in Fig. 22.5 it becomes obvious how the before mentioned classification can help with solving certain problems by traversing a graph and evaluating the data collected along the found path.

Taking up the afore mentioned example of the wished analyzation of a certain cooling circle, the following question arises:

Question: Are there any current or upcoming problems in cooling cycle X?

In order to answer this question a path through the graph of the cooling cycle X needs to be found via an algorithm. Once it is found the path will be iterated front to back, while on each vertex relevant neighboring vertices connected through selected edges of the right classes are analyzed.
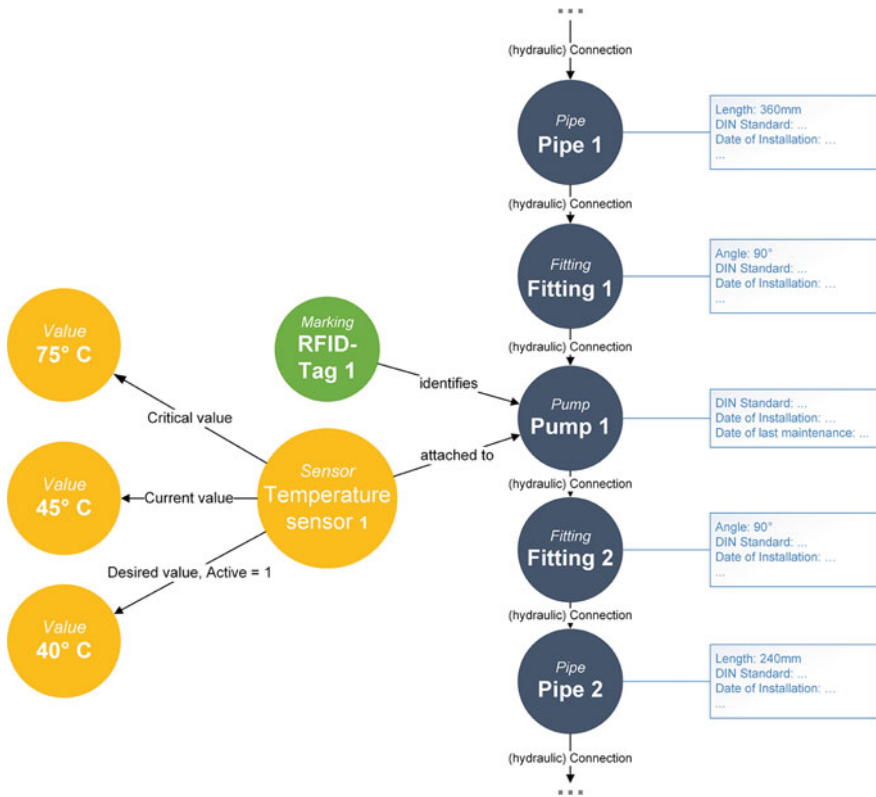
Fig. 22.5 Exemplary excerpt from the graph model

Applied to the example in Fig. 22.5 one can see that each vertex has stored data and connections to other vertices. The algorithm starts off and finds a path that includes the section between *Pipe 1* and *Pipe 2*. After the successful path-finding, the found vertices are iterated and checked against predefined functions to evaluate their current status. *Pump 1* is equipped with the sensor *Temperature Sensor 1* which is connected to value vertices. The values in itself hold data about a certain value and the value's measuring unit. Through this process each sensor is processed individually and offers maximum flexibility. The information about the value's significance is expressed in the type of connection between the sensor and a value. In this example *Temperature Sensor 1* offers information to the system which basically translates to the answer of the initial question:

Answer: The temperature of Pump 1 is higher than desired but within the tolerance range.

To get this process to create work instruction or indications for the technician in the field, it is not necessary to create step-by-step instructions for instructions concerning sensor-equipped elements, but to create *Jobs* with *Tasks* that include
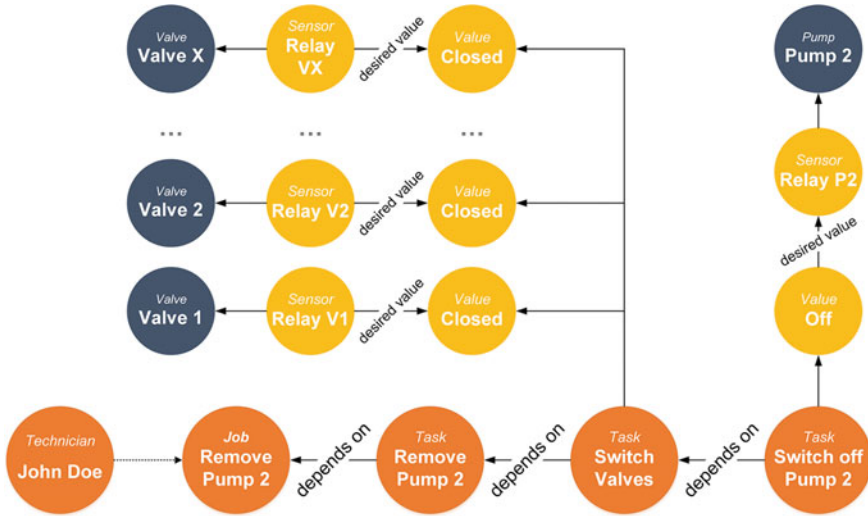
Fig. 22.6 Exemplary job for automated instruction generation

dependencies between the tasks and the desired *Values* of sensors concerned in the planned maintenance process. To elaborate this, Fig. 22.6 shows an example where two tasks are defined with values that need to be in place before the task is completed. The first step in this *Job* to remove *Pump 2* is to switch off the relay controlling the pump. No work instruction would be created if the desired status was equal to the current one. Same goes for the status of valves in the follow-up task. Only for non-closed valves would an instruction be generated.

The desired values which are connected through the according edge class can specify a criticality level based on ISO 26262's hazard analysis and risk assessment. The resulting criticality levels in the concept are defined as follows (S = Severity).

S0: No damage, no functional components affected
S1: Light damage, reduction of efficiency
S2: Hazardous damage, loss of functional components
S3: Catastrophic damage, loss of the safety related components or complete machine

Although it is in the nature of such a concept to capture and evaluate data from multiple (conceptually from an unlimited number of) sensors, we do not implement an actual multi-sensor data fusion. Hall [25] states that a multi-sensor data fusion is characterized by statistical advantage gained by combining same-source data or increased accuracy trough use of multiple types of sensors.

In the current state of this concept, the focus lies on the dynamic evaluation of the captured data, based on functions and algorithms that offer direct evaluation.

As stated in the requirements section, the data providing modules are to be made available as a cloud-solution via internet connection. The *Data Provision* layer

thereby consists of a set of web service providing components to communicate with the on-site clients, as well as other systems on a remote location.

22.3.3 Maintenance Assistant

The maintenance assistant application provides the user interface for technicians. The *Maintenance Assistant* is conceived as an application for mobile devices and supports the maintenance personnel on-site with data provided by the *Data Analysis* layer.

From here the technicians can access all offered functions for their specific task. The functionalities of the application can be divided into two categories, first the *Data Usage*, second the *Data Generation*.

The authors already published a description of a graph-based maintenance assistant system for smart devices in [13]. The then developed concept for an augmented reality application must be adjusted to match the new circumstances, for instance the pre-known current machinery status through implementation of IoT components and the confrontation with possible a ban on photography.

In order to use an augmented reality application which obviously is dependent on its camera various arrangements have to be made. Components of an augmented reality application are arranged in the categories *display, tracking and interaction* [26]. The virtual part of the augmented *display* principally consists of digital 3D-models. To increase performance on mobile devices, the general approach is to reduce 3D models to mock-ups, that still feature the same overall geometry, but are far less complex.

Tracking in case of augmented reality denotes location determination of user (or the displaying device) as well as from distinctive characteristics and objects in space. This can be achieved via various procedures which the authors [27] looked upon in an earlier publication. The possibilities of interactions in the AR-environment are versatile and span from marker-based input, motion tracking or simple touch gestures on the display of the smart device [26].

In Chap. 2 the reality-virtuality continuum was discussed along with the possibilities to use AR application components to build a camera-independent 3D application. Starting again at the core components of an augmented reality application (display, tracking and interaction) the guidelines for the development of the 3D mode can get designed.

As already stated, the 3D mock-up models are presented as an overlay over the camera picture within an AR application. The same models can be displayed from within the 3D engine that renders the 3D models with an artificial background instead of a camera picture.

As for the tracking component it is obvious that without the camera and for that regard similar systems like infrared or laser based measurement systems, it is not possible to optically track the relative position between the user/the user's smart device and the concerning machine. For the initial pose a small set of data can be

stored right on a RFID tag on the machine to align the model. This data set can be encoded in JSON and stored as regular text on the tag to be used on every RFID-enabled device. An example for such a data set matches the following pattern to provide the application with a unique identification, a human-readable description and the initial roll-pitch-yaw angle:

```
{

"ID": "1234",
"Label": "Pump 1",
"RPY-Angle": [15, 0, 90]

}
```

Instead of using the positions of markers or inherent features calculated with the AR-framework, a gyroscope which is usually fitted in every smart device nowadays can be used to keep the 3D model in roughly the right angle to the user after establishing the initial pose.

Interaction will be implemented via gestures in order to steer the 3D-rotation, as is the standard in modern mobile devices. Basically the same principles apply to both viewing modes, so that the *Data Usage* perceived as representation of the guidance is identical from a data perspective.

The concept's top layer *Data Generation* is responsible for creating the feedback data that is brought back to the source systems. This data may consist of log files of fulfilled tasks, used material, exact timing, involved personnel and parts and manual entries from the technicians.

Implementation targets for the maintenance assistant application are industrial mobile devices, generally in the category of tablets. Given that the tablet at hand is equipped with a camera, the maintenance assistant will be able to switch to either 3D or AR mode at the beginning of a maintenance process.

22.4 Prototypical Implementation and Validation

For the demonstration of validity of the presented approach our validation is centered around the complex hydraulic system (Fig. 22.7) which was also used in the case study in [12].

The hydraulic system has two centrifugal pumps, two tanks and twelve valves. The construction of the hydraulic system allows the fluid to be pumped through both of the two pumps and through each of the two containers, as well as through both pumps in a row. The respective fluid circuit depends on the combination of the respective valve positions.

The hydraulic system was upgraded with newly designed sensors to gather the status of the valves and pumps automatically. Sensor modules were designed and manufactured which gather the lever positions of the valves and so distinguish
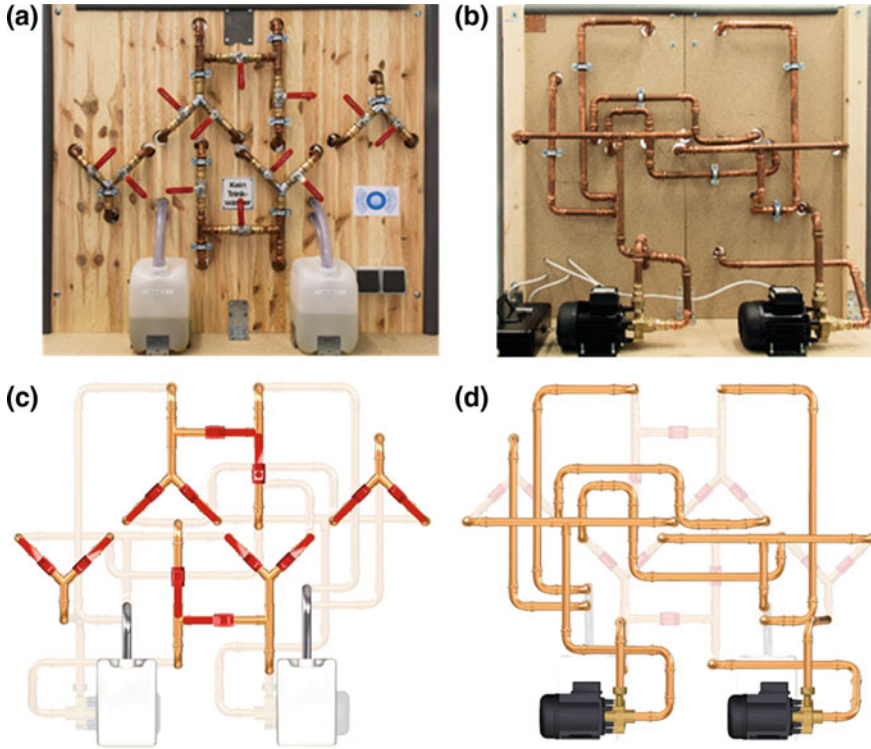
Fig. 22.7 Front (**a**, **c**) and backside (**b**, **d**) of the hydraulic system in both reality and virtuality

between open or closed lever positions as well as capturing the unsafe state in between. Furthermore, the installed pumps have been wired via relays which are switchable and readable by the controlling Raspberry Pi 2 on the basis of self-developed web service-protocol. With this the sensor data of the hydraulic demonstrator is accessible for the data sourcing module via the mentioned interface.

The general structure of the concept (cf. Fig. 22.2) has been preserved in the implementation (cf. Fig. 22.8). For easier understanding the functional units were remodeled after the distribution of functionalities offered by self-developed components as well as commercial software.

The software products used were:

- Data Sourcing

    – Data Acquisition: PTC Windchill, Self-developed components
    – Data Consolidation: PTC ThingWorx, Self-developed components

- Data Handling

    – Data Analysis: PTC ThingWorx, OrientDB, Self-developed components
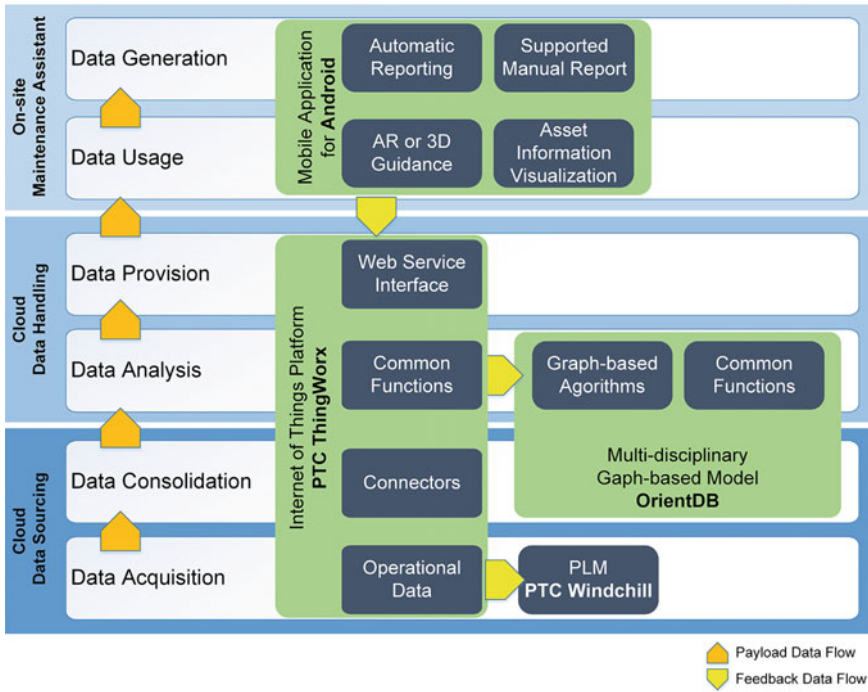   – Data Provision: PTC ThingWorx, Self-developed components

Fig. 22.8 Implemented system architecture

- Maintenance Assistant

    - Data Usage: PTC Vuforia, Google Android, Self-developed components
    - Data Generation: PTC Vuforia, Google Android, Self-developed component

Generally, a loose coupling of components on web services of XML- or JSON basis was implemented in the prototype in order to achieve maximum flexibility. These technologies are generally considered to be feasible for the use in IoT applications [19].

Two major components are relevant in this module of the implementation. First there is PTC Windchill as a source system for 3D models and meta data, secondly there is PTC Thingworx as the central point of communication between the different components and product instance specific data.

The 3D design data was deposited in PTC Windchill for later use. With help of the so called connector, Windchill-data can be accessed on the IoT-platform PTC ThingWorx. In addition, ThingWorx is capable of seizing precast meteorological services and retrieve these for each specific GPS-coordinate provided.

The hydraulic system possesses a digital twin [28] in PTC ThingWorx, which means every set of data referring to the special product instance (operating data and for example spare parts etc.) are administrated here.

OrientDB is a graph database with document oriented store option [28] and in this implementation utilizes predefined a conceptual graph database schema according to the concept described in Sect. 3.2 instead of an automated import from i.e. the product structure. Within the context of the prototypical implementation this work was done by hand in order to examine the functioning and evaluate the dynamic extensibility rather than laying focus on the import process from existing models. The import process and the resulting automated model building will be a subject in future research.

The administration of product instance data takes place in the PTC ThingWorx, while the administration of the semantic and operational data is organized in OrientDB.

The planned functions for the graph traversing and analyzation were deposited in the graph database management system as java script user functions and made accessible for other components via the build in PHP-web service-implementation of OrientDB.

The *Maintenance Assistant* application was implemented with PTC Vuforia and the open source 3D-engine JPCT-ae. As mentioned in the concept, the application supports two modes of usage to present either augmented reality or 3D content.

To show the capability of the approach at hand we prepared two use cases in laboratory environment. The first being the simple monitoring of operational data, the second being the maintenance assistance with context-sensitive indications.

22.4.1   Visualization of Real-Time Operational Data

The first implemented operation mode shows selected operational data or to be more precise analyzation reports based of operational data. The implemented functions for data evaluation have an immediate impact on the options the technician can choose from on-site. The following options were implemented to choose sensors based on

1. their type,
2. their unique identification,
3. the family of parts they are attached to,
4. their general position on the machine or
5. the parts that are relevant in the assigned task.

These modes generally filter the multitude of managed sensors in the graph-based model. As shown in Sect. 3.2 sensors are connected to their specific boundary values individually while they are analyzed according to their edge classification. Further filtering can be applied by selecting a minimum criticality level for the displayed indications. As an example only "*category D: critical condition*" evaluated indications concerning the front side of the hydraulic system
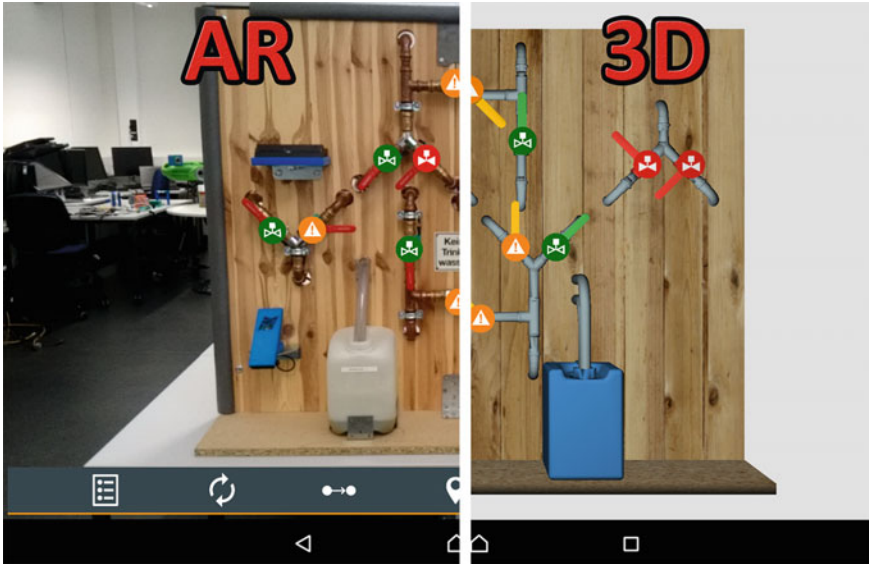
Fig. 22.9 Augmented reality visualization (*left*) and 3D visualization (*right*) for warnings

are shown in Fig. 22.9. The center value on the very top and bottom of the machine are in an unknown state, which causes the functions to report a critical condition.

If a change of state occurs (i.e. by switching a valve, which triggers updates in the graph based model) and exceeds a critical limit in the monitored system, the particular technician immediately gets automated warnings and indications concerning the current context. If the change brings the system into an uncritical state, only minor indicators are updated.

22.4.2 Context-Sensitive Maintenance Assistance

The second use case is comparable with the implementation the authors conducted in [13]. The maintenance assistance with context-sensitive work instructions was implemented based on the assumption that *pump 2* needs to be removed (cf. Fig. 22.6) for later replacement. For that matter *Pump 1* needs to handle the flow through both tanks in the system.

This premises induce certain desired conditions which then are evaluated considering the machines current state to generate the required instructions for the technician.

Figure 22.10 show the resulting instructions and indications. The list to the left of the screenshots offers a quick overview on the correct and incorrect valve positions for the working step "switch valves" in Fig. 22.6 in reference to the systems status at that time.
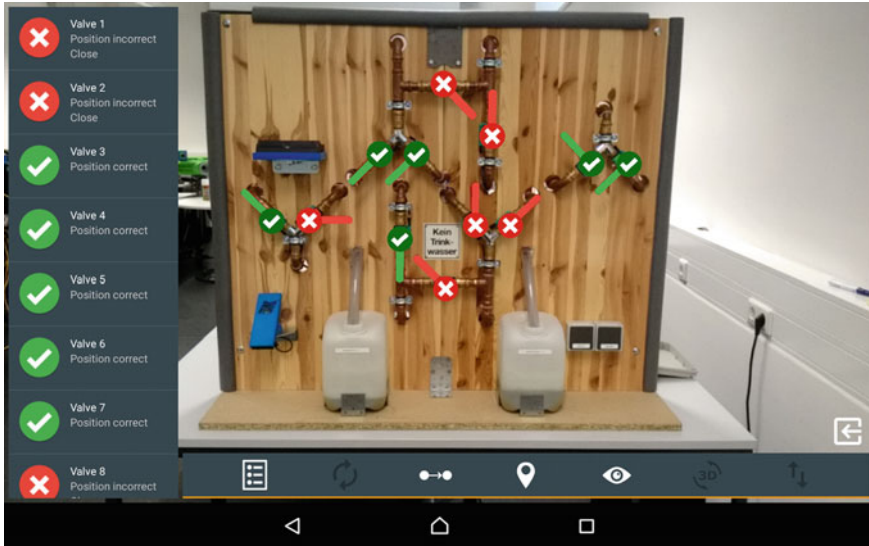
Fig. 22.10 Augmented reality visualization for maintenance guidance

It is easy to see that while the augmented reality mode (Fig. 22.9, Left side) offers the intuitive viewing of instructions and machinery alike, the amount of information shown is identical to that of the camera-less 3D mode (Fig. 22.9, right side).

22.5 Conclusion and Future Work

The paper at hand exhibits the potentialities of the graph-based approach for context sensitive analysis of company-wide data and the direct inclusion of real-time sensor data on-site. The highly dynamic graph-based data model offers the flexibility needed for adapting to changes in the operation of smart, agile manufacturing scenarios.

As shown in the use case scenario the approach ensures that only necessary steps are taken to reach a certain desirable machine status.

The approaches presented in previous publications have been considerably enhanced and the resulting concept presented in this paper has been validated in a laboratory experiment. The lessons learned result in several improvements for the implemented prototype, such as:

- Direct interaction with different source systems other than the IoT platform
- Offering more support to the technician, if need be. The amount and type are to be determined by a survey in the near future
- Adding the capability to visualize the current work status of a certain technician on a remote PC in a web-based client

The future research activity will focus on enhancing the presented approach further than the obvious improvements mentioned above.

To achieve a consistent process that expands into the earlier development phases of products, an import process must be developed to "feed" the graph database with its necessary data. To this extend the import of product structure data is planned, as well as implementing interfaces to synchronize data between the graph-model and a CMMS.

As the task management is already implemented for a single technician, the task management concept will be revised to support on-site collaboration between multiple technicians working on the same facility on large-scale repair or maintenance jobs. The collaboration module will track and coordinate all safety-related stages, and issue coordination instructions when reaching synchronization or combination points in the combined workflow of the technician team.

Taking one step further from the augmented virtuality, the idea of tracking and reviewing the technician's information from afar, the concept will be brought to the final stage in the reality continuum (cf. Fig. 22.1, [21]), virtual reality. As the prototypical implementation is already able to display 3D visualization of the indications and the concerned asset, only the immersion is left to be sorted out to realize a virtual reality counterpart of the real work for a remote technician to review.

In preparation for the next evolutionary step of the presented concept, we will adapt to a more complex demonstrator machine that features pneumatics, electronics, PLC control, RFID-based sensors and is usually used in professional education of technical personnel.

References

1. Aleksy M, Rissanen MJ, Maczey S et al (2011) Wearable computing in industrial service applications. Procedia Comput Sci 5:394–400. doi:10.1016/j.procs.2011.07.051
2. van Krevelen DWF, Poelman R (2010) A survey of augmented reality technologies, applications and limitations. Int J Virtual Real 9(2):1–20
3. Dini G, Mura MD (2015) Application of augmented reality techniques in through-life engineering services. Procedia CIRP 38:14–23. doi:10.1016/j.procir.2015.07.044
4. Schlick CM, Jeske T, Mütze-Niewöhner S (2014) Unterstützung von Lernprozessen bei Montageaufgaben. In: Schlick CM, Moser K, Schenk M (eds) Flexible Produktionskapazität innovativ managen: Handlungsempfehlungen für die flexible Gestaltung von Produktionssystemen in kleinen und mittleren Unternehmen. Springer Vieweg, Berlin
5. Abramovici M, Göbel JC, Neges M (2015) Smart engineering as enabler for the 4th industrial revolution. In: Fathi M (ed) Integrated systems: innovations and applications. Springer International Publishing, Cham, pp 163–170
6. Reinhart G, Patron C (2003) Integrating augmented reality in the assembly domain—fundamentals, benefits and applications. CIRP Ann Manuf Technol 52(1):5–8. doi:10.1016/S0007-8506(07)60517-4
7. Eisenhauer M, Oppermann R, Prinz W (2007) Internet der Dinge: Anwendung von RFID- und Tracking-Technologien zur intelligenten kooperativen Assistenz im Arbeitsprozess. In: Internet der Dinge. Springer, Berlin, pp 49–62. www.internet-der-dinge.de

8. Wohlgemuth W, Friedrich W (2007) Advanced augmented reality technologies for industrial service applications: ARTESAS; Sachbericht Gesamtvorhaben; zusammenfassender Schlussbericht; BMBF-Leitprojekt ARTESAS; Laufzeit/Berichtszeitraum: 01.02.2004–30.06.2006; Berichtstermin: 30.09.2006, Version: 30.01.2007. Siemens AG, Nürnberg
9. Barthelmey A, Störkle D, Kuhlenkötter B et al (2014) Cyber physical systems for life cycle continuous technical documentation of manufacturing facilities. Procedia CIRP 17:207–211. doi:10.1016/j.procir.2014.01.050
10. Hou L, Wang X, Truijens M (2013) Using augmented reality to facilitate piping assembly: an experiment-based evaluation. J Comput Civ Eng 29(1):5014007. doi:10.1061/(ASCE)CP.1943-5487.0000344
11. Azuma RT (1997) A survey of augmented reality. Presence: Teleoper Virtual Environ 6 (4):355–385. doi:10.1162/pres.1997.6.4.355
12. Abramovici M, Krebs A, Wolf M (2014) Approach to ubiquitous support for maintenance and repair jobs utilizing smart devices. In: Horváth I (ed) Tools and methods of competitive engineering: digital proceedings of the tenth international symposium on tools and methods of competitive engineering—TMCE 2014, May 19–23, Budapest, Hungary. Faculty of Industrial Design Engineering Delft University of Technology, Delft, pp 83–93
13. Neges M, Wolf M, Abramovici M (2015) Secure access augmented reality solution for mobile maintenance support utilizing condition-oriented work instructions. Procedia CIRP 38:58–62. doi:10.1016/j.procir.2015.08.036
14. Partynski D, Koo SG Integration of smart sensor networks into Internet of Things: challenges and applications. In: 2013 IEEE international conference on green computing and communications (GreenCom) and IEEE Internet of Things (iThings) and IEEE cyber, physical and social computing (CPSCom), pp 1162–1167
15. Finkenzeller K, Gebhart M (2015) RFID-Handbuch: Grundlagen und praktische Anwendungen von Transpondern, kontaktlosen Chipkarten und NFC, 7, aktualisierte und erw. Aufl. Hanser, München
16. Schulz D, Gitzel R Seamless maintenance—integration of FDI device management and CMMS. In: 2013 IEEE 18th conference on emerging technologies and factory automation (ETFA), pp 1–7
17. Sean W, Levi L, Steven F (2007) Visual hints for tangible gestures in augmented reality. In: 6th IEEE and ACM international symposium on mixed and augmented reality, 2007, ISMAR 2007, pp 47–50
18. Abramovici M, Krebs A, Lindner A (2013) Exploiting service data of similar product items for the development of improved product generations by using smart input devices. In: Abramovici M, Stark R (eds) Smart Product Engineering. Springer, Berlin, pp 357–366
19. Andrea Z, Nicola B, Angelo C et al (2014) Internet of things for smart cities. IEEE Internet Things J 1(1):22–32. doi:10.1109/JIOT.2014.2306328
20. Dörner R, Broll W, Grimm P et al (eds) (2013) Virtual und augmented reality (VR/AR): Grundlagen und Methoden der Virtuellen und Augmentierten Realität. eXamen.press. Springer Vieweg, Berlin
21. Milgram P, Takemura H, Utsumi A et al (1995) Augmented reality: a class of displays on the reality-virtuality continuum. In: Das H (ed) Photonics for industrial applications. SPIE, pp 282–292
22. Haberfellner R, Daenzer WF (eds) (2002) Systems engineering: Methodik und Praxis, 11., durchges. Aufl. Verl. Industrielle Organisation, Zürich
23. Harrison G (2015) Next generation databases: NoSQL, NewSQL, and Big Data: what every professional needs to know about the future of databases in a world of NoSQL and Big Data. Apress (IOUG), New York, The expert's voice in Oracle
24. Vajna S, Bley H, Hehenberger P et al (2009) CAx für Ingenieure: Eine praxisbezogene Einführung, 2, völlig neu, bearb edn. Springer, Berlin
25. Llinas J, Hall DL (1998) An introduction to multi-sensor data fusion. In: ISCAS '98 1998 IEEE international symposium on circuits and systems, pp 537–540

26. Tönnis M (2010) Augmented reality: Einblicke in die Erweiterte Realität. Informatik im Fokus, vol 0. Springer, Berlin
27. Koch C, Neges M, König M et al (2014) Natural markers for augmented reality-based indoor navigation and facility maintenance. Autom Constr 48:18–30. doi:10.1016/j.autcon.2014.08.009
28. Rosen R, von Wichert G, Lo G et al (2015) About the importance of autonomy and digital twins for the future of manufacturing. IFAC-PapersOnLine 48(3):567–572. doi:10.1016/j.ifacol.2015.06.141