Applying VR, AR, OCR, and 3D Feature Tracking for Automotive Maintenance Tasks

Renato Turić, Richard Hable
Evolaris next level GmbH
Hugo-Wolf-Gasse 8-8a, 8010 Graz, Austria
{renato.turic, richard.hable}@evolaris.net

Gerhard Brugger
Porsche Informatik. Gesellschaft m. b. H
Louise-Piëch-Straße 9, 5020 Salzburg, Austria
gerhard.brugger@porscheinformatik.at

Abstract. *In this paper a new way of simplifying automotive maintenance tasks is proposed. It uses state of the art augmented reality (AR) for 3D tracking, optical charachter recognition (OCR) for identifying the types and models of cars, and virtual reality (VR) for the visualization of damages for service technicians. Also, special hardware capabilities like the motion sensors of the Google Glass product are applied, supporting hands-free operation on this head-mounted device. An application has been created which gives the service technician all the information needed without technology interfering with the actual work of repairing a vehicle. Also, the amount of paper flow during the maintenance process is reduced. Based on design science principles the suitability of the created artifacts for the defined application field is evaluated.*

Keywords. augmented reality; virtual reality; optical character recognition; Google Glass; automotive maintenance; 3D tracking; wearables

1 Introduction

This paper is the result of a project that was created for helping a service technician to get relevant information about damage reports of a specific car. The damage reports are data based on the company's central server infrastructure. The damages on a car can vary between scratches, damages from rock falls, any kind of bumps or dents and other damages.

One of the main goals for this project was to create a hands-free experience for the service technician while overviewing the damages of a specific car so that the technology is not intervening with his actual work of repairing the car. This is where we decided to use a wearable device for the application deployment. There are several types of wearable devices like watches, necklaces, glasses, bracelets, shirts and so on. This project is concentrated on smart glasses only.

The application developed in this project is able to recognize the specific car that is currently on the list for the repair process and based on that information to deploy the damage reports to the wearable device. The damage reports show detail descriptions of the damage and also visualize these damages on a model through virtual reality and augmented reality. The virtual reality part helps the service technician visualize all the damages of the car in one place and get a quick overview where the damages are. The augmented reality visualizes one specific damage and gives the service technician information about the damage, providing him/her with hints how to fix the damage.

2 State of the Art

The field of car maintenance is usually very conventional in the aspect of technology and paperwork. The process is normally consisted of several steps. The first step is the assessment of the damage to the vehicle by the customer and the agent of the insurance company. The damage assessment document is normally sent to the repair site, and a service technician is assigned. The service technician needs to analyze the damage assessment and order necessary replacement parts. Normally, there are quality inspections of the workmanship between some stages of the repair process to check if everything is working as it should be. When finished the service technician notifies the customer and/or the agent and prepares the bill.

New technologies are emerging both in the software and hardware areas. Wearable devices provide features and applications which are not available on stationary devices. Especially, smart glasses are getting more and more popular. Considering the fact that most smart glasses have hardware cameras and enough computing power to render 3D objects, developers can use augmented reality to a more reasonable level than on the tablet or web-cam.

An example application that uses the AR and wearable concept in the area of car maintenance is Metaio's first hands-free car manual application on Google Glass. They presented a demo of the application on the InsideAR 2013 conference in Munich [15]. The application is using the state of the art 3D tracking for the augmentation process which we also used in this project.

The smart devices and the AR concept can be exploited for educational purposes too. An example project is an e-training mobile application for car maintenance training which uses augmented reality that supports real-time content update when the content is modified (CDN) [8].

More and more wearable devices are heavily oriented for augmented reality. Also, some older technologies like Kinect can reproduce augmented reality that involves not only the vision of the user but also body gestures. Combining these two features we can get to some innovative solutions, e.g. a group of engineers recently developed a document browsing method based on a book-style interface usable in a normal office environment using augmented reality and Kinect [9].

3 Use Cases and Requirements

We want to deliver necessary information to the service technician for the process of repairing a car. Other roles beside the service technician are the agent and the customer. Those roles are not the main interests for this project but they are important for understanding the concept of the project. The customer helps the agent, and vice-versa, to recognize all the damages of the customer's car. The damages are stored in a database on the server and can be later downloaded and shown on the wearable device. After the data is uploaded on the server and all the content is created for the virtual and augmented reality, it is ready for the service technician to use.

The following chapters show how the project is divided into use cases and the main purpose of each use case.

3.1 UC1 - Recognize the car

The first task is to determine the car that is needed for the process of downloading the prepared data. In this case the license plate is the perfect unique identifier between cars. The license plate should be scanned in a manner that is most suitable for the wearable device. The best way to get that information and have the hands-free experience is to use optical character recognition. Another option is to use voice dictation. In our case we included both functionalities, the OCR as the main functionality and the voice dictation as the fallback solution.

After the scan or dictation is completed and successful, the process of downloading all the necessary data from the database begins. If the license plate number is in the database the download will be successful (overlooking internet communication problems) and the wearable device will receive the needed information. This leads to the second use case.

3.2 UC2 - Fetch damage information according to license number

In this use case the implementation of the downloading process of the data from the database is implemented. This database represents the small backend part of this project and it is not in the focus of the project, as we concentrated on the presentation of data on the wearable device and helping the service technician in his work. The only requirement for this use case is to have a simple and flexible structure that is easy to use.

After receiving the data from the backend, the next use case starts.

3.3 UC3 - Present reported damages

In this use case the data (i.e. damage reports) is presented in a way that is also easy readable and intuitive for the service technician on the wearable device. After the download of the data is finished, the damage reports are shown in two different ways. One way is to show all damage reports in a virtual reality manner where the service technician can use the sensors of the device to manipulate the object, in this case a car with marked damages. The other way is to show the damage reports in a simple list. From the list the service technician can choose a damage report to get detail information about it. Once the service technician selects one damage report the next use case follows.

3.4 UC4 - Present single damage

Selecting a single damage report leads to a list of information about that specific damage. The information is also downloaded from the server and contains the cause and the position of the damage, possibly including hints on how to fix it or what to be careful about. The detailed information should be simple and clear.

From this screen it is possible to open a menu for choosing between seeing the photos of the damage and seeing the augmentation of the damage on the real car while walking around it.

3.4.1 Show photos of the damage

When selecting the menu option to see photos of the specific damage the service technician can choose between different photos of the damage on the car. The photos are taken from the database on the server and are shown on the device's screen in a manner that is clear and obvious for the service technician.

3.4.2 Augmentation of the damage

When selecting the menu options to see the augmentation, the wearable device gets all needed data from the server and the actual augmented reality experience starts.

4 Design Method

Trying out new technology, especially in areas like augmented reality, where the capabilities of hardware and software increase within months, requires experimentation and the creation and evaluation of prototypical artifacts. Therefore, we decided to base our project on design science principles [7]. Thus, even though a comprehensive solution for a concrete application field was created, the implemented functionality was mainly driven by the need to create a usable implementation suitable for scientific evaluations. When different solution paths showed different strengths and weaknesses, concurrent implementations were created which can now be used alternatively. On the other hand, no effort was made to find new solutions for already well-established areas like managing customer data on centralized server systems. The following descriptions show how the seven design science guidelines described in [6] were applied:

Design as an Artifact: Working implementations were created with newest available hardware and with software partly still in a beta state. This led us to solutions of different practical usability. For example, 3D tracking based on CAD data turned out to be rather cumbersome due to the slow and complicated initialization process necessary. Nevertheless, the results provided valuable information about the current state of the art, and the situation will certainly improve in the future due to progress in processing power and framework quality.

Problem Relevance: The cooperation with a commercial company in the chosen application field lead to solutions for actual business problems. The artifacts work with data based on actual repair data structures used on the company's central server infrastructure, and tasks are supported which are performed on an every-day basis in practice.

Design Evaluation: The created artifacts have been tested for applicability on clearly specified practical tasks. Even though the resulting application is not intended for daily practical use, it is already possible to compare how well working steps of service technicians are supported either by conventional solutions or by the new methods implemented in the designed artifacts.

Research Contributions: New technology is applied to support practical tasks in a clearly specified application field. This shows the current implementation state of the art and allows an estimation of promising future research and application areas.

Research Rigor: All artifacts have been created based on realistic data and deployment architectures. Models of real cars are used for 3D tracking and augmentation, and a realistic client-server architecture including the transfer of significant amounts of data through the Internet has been implemented. This way it is possible to evaluate the real behavior of the application in a practical setting, including network delays and limitations due to limited processing power and memory constraints.

Design as a Search Process: Design of the application started with an exploration of the currently available technology and then continued with practical software implementations checking the achievable quality of different solutions. When an implementation did not turn out to be satisfactory, alternative tools and methods were searched. For example, when text recognition of license plates in realistic surroundings turned out to be too unreliable, printed license plate stickers were introduced, and even voice recognition was built in as a different input method.

Communication of Research: The following chapters describe the designed artifacts, including experiences gained during software implementation and practical evaluation. This will allow other developers to make informed decisions about their own implementations and help management to estimate the applicability of the described new technologies in their own application areas.

5 Designed Artifacts

Augmented reality on handheld systems is an area that is already widely explored, and taxonomy categorizing research and development in handheld augmented reality applications (HAR) is available [5]. This is not the case for augmented reality for wearables and hands-free applications.

5.1 UC4 - 5.1. Hands-free solutions

In our project we chose Google Glass from several wearable devices. We made a comparison between the Google Glass, Vuzix M100 and Epson Moverio BT200. During the decision process it was important to look at aspects like screen resolution, see-through view, and form factor (monocular/binocular). Table 1. provides an overview of the comparison results.

Table 1. Smart glasses comparison table

Smart glasses feature	Google Glass	Vuzix M100	Epson Moverio BT20
See-through	Yes	No	Yes
External control panel	No	No	Yes
Form-factor	Monocular	Monocular	Binocular
Resolution	640x360	240x400	960x540
Field of view	14°	15°	23°
Weight (grams)	50	Not published	220
CPU	Not published	OMAP4460 1.2GHz dual core	1.2GHz dual core
RAM	Not published	1GB	1GB
Connectivity	Wi-Fi 802.11b/g/n	Wi-Fi 802.11b/g/n	Wi-Fi 802.11b/g/n
Bluetooth	Yes (4.0 LE)	Yes (4.0)	Yes (3.0)

The overall result of the comparison is in favor for the Epson Moverio BT200 but we nevertheless chose Google Glass for this project. The main reason why we favored Google Glass over Moverio in this case is the required control panel and the weight of the glasses. The wire connection between the glasses and the touchpad for input is just not suitable for a project where hands-free is one of the focus points. The main difference between Google Glass and Vuzix is the missing see-through feature on the Vuzix. Also the low screen resolution on the Vuzix is a disadvantage for augmented reality solutions.

5.2 Innovative input methods

The application supports several input methods, depending on the activity. Sensor based navigation is used to select menu items and to find damage positions on a displayed car model. This allows exploring the advantages of hands-free usage. In some cases the service technician needs to do some inputs with fingers; taps and swipes. It would have been possible to implement a wink gesture as an alternative to taps, but for the cost of simplicity and reliability.

Optical character recognition (OCR) together with smart glasses is a nice fit but it can easily backfire if the implementation is not sufficiently reliable, which is discussed in detail later on. In our project we chose Tesseract OCR as the engine for text recognition. It is an open-source OCR engine that was developed at HP between 1984 and 1994 and now the property of Google Inc. [11]. As an OCR engine it is behind the leading commercial engines in terms of its accuracy but still good enough, performance-wise, for this project.

Speech recognition is implemented as a fallback solution if the service technician has problems with getting the text recognition to work properly. The built-in voice dictation in the Glass Development Kit (GDK) is well suited for capturing spoken letters and numbers of license plates.

5.3 Virtual and augmented reality

The described project contains both a virtual and augmented reality aspect. The virtual reality is implemented with a 3D digital object of a car with marked damages and the usage of sensors from the device. This gives the service technician the ability to interact with the object using those sensors and simulate walking around the car or lifting it up on a car lift to see it from bellow, or even climbing on a ladder to see it from above. Using the hands-free navigation system the service technician can look at all the damages in real time without going near the car. Fig. 1. shows screenshots of the virtual reality view.

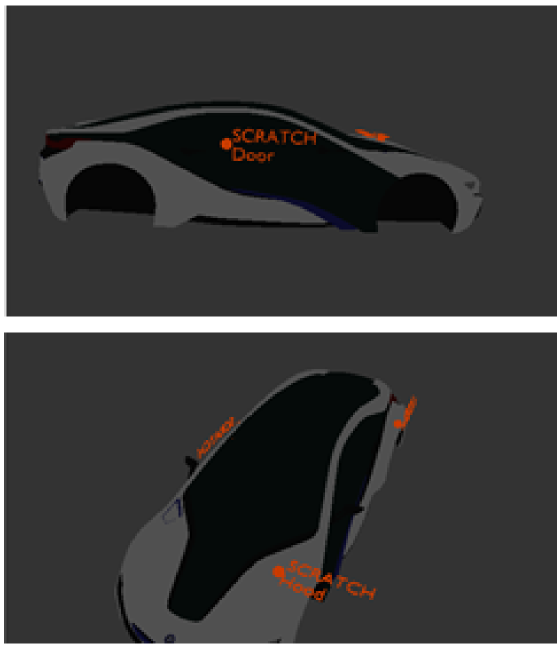

Figure 1. Virtual reality views

The augmentation aspect is implemented by using the 3D tracking of the actual car and showing useful information about a specific damage. In this project we use the Metaio framework for augmented reality and 3D tracking. Tracking can be achieved with many different types of sensors, including GPS, gyroscope, camera, hybrid vision, accelerometer, as reported in [3] [2] [4] [9]. Metaio offers two types of markerless tracking that are suitable for our project: 3D tracking based on CAD data and tracking based on 3D maps (object and environment tracking). Fig. 2. shows a screenshot of the augmented reality view using a car model.

Figure 2. Augmented reality aspect of the project (augmented content on the Google Glass)

5.3.1 3D tracking based on CAD data

In this project 3D tracking based on CAD data is preferred because it provides reliable result and does not depend on the color of a car or the lighting of the environment. For this 3D tracking to work a line model, also known as an edge based model, and a surface model are needed. Depending on the type of tracking there are corresponding configurations for each individual tracking option, like fixed view indoor, free view indoor or free view outdoor [13]. It is important to note that this feature is still in development (beta phase) and does not work well on low-end devices.

5.3.2 3D maps based tracking

Creating a 3D map for the Metaio platform is done with the Toolbox program. The created 3D maps consist of point clouds and key frames. It does not track edges like the 3D tracking based on CAD data. Instead it tracks the point cloud features recorded during the creation of the 3D map, therefore depending heavily on the quality of this map [14].

6 Evaluation

The project involved Google Glass, Metaio (3D tracking based on CAD data), the Tesseract-OCR engine, OpenGL and a free 3D solution for Java and Android called jPCT 3D engine. Some work was also done in Blender and Unity. Unity was mainly used for testing the 3D tracking before the implementation on the Google Glass. The following chapters contain a first evaluation of the artifacts based on this hardware and software.

6.1 Google Glass

The smart glasses from Google are not well suited for applications which use a lot of computing power due to overheating and low battery capacity.

The small screen demands a simplistic user interface. It is recommended to avoid complex graphics and to simplify the implementation in order to use less computing power. That is why we had to simplify all our models. Augmented content had to be optimized for the Glass, to be minimalistic, short and brief.

Glass has 12 GB of usable memory (16 GB Flash total), which was more than enough for the application of this project. It was mandatory to optimize the models and remove all unnecessary effects to reduce the size of the models.

6.2 Figures

Besides the Metaio platform for augmented reality we looked into the Qualcomm Vuforia platform. During research Metaio turned out to be the preferable platform because of its 3D object tracking. Vuforia offers a similar feature, but it only tracks objects image by image, and is very limited concerning the allowed complexity of the object [12].

It was necessary to research the technology in detail and to do some testing for the 3D tracking based on CAD data. The early 3D tracking samples on a Nexus 10 tablet provided some good results. One of the results, however, showed that Metaio is using a very big amount of computing power for the occlusion edges of a model. Therefore, when creating a model it is important to check the polygon number of the mesh, and detect lines that are best suitable for detectable edges. All the surfaces of the models need to be triangulated, and while exporting the models, the Y-axis should always point forward and the Z-axis should point upwards [13]. It turned out to be difficult to find or create models simplified that way.

First tracking results using our simplified car model were unsatisfying and demanded further optimization of the models and the configuration files for the tracking. The main goal of the optimization was to lower lag issues while initializing the tracking, reduce overheating of the device, and to limit the battery consumption.

6.3 Tesseract-OCR

The optical scanning of the characters on the license plate is a resource demanding feature on the Google Glass. This is the reason why the OCR is better done by scanning a sticker of the license plate instead of the actual license plate. Recognizing black text on a white background requires a lot less resources than using image segmentation of the real license plate with computer vision technology [1]. Then it is also not necessary to train the Tesseract engine for a specific font of the target license plate.

6.4 Usability and practical use

Designing the application's user interface required analyzing the current state of the service workflow. Different areas were identified in which manual processes could be replaced or at least improved with better tooling. Therefore, the application was designed to span not only a single working task but to help in the complete process of recognizing damages up to actual repairing. This includes paper work which is currently usually performed manually, but can more easily be done using automatic data collection and processing. But, of course, in order to explore new technology in the project, the main focus was on innovative usage of the new mobile devices.

Therefore, state of the art technologies were applied to give the service technician a working environment with less paper work and with hands-free repairing. The virtual reality also gives the service technician the ability to see all the damages without the need to lift the car. Augmentations of the damages give the service technician a brief overview of the damage that is useful for his repair process, including the type of replacement part to order.

Following design science principles, a design evaluation was performed on several sample repair processes. It showed that with application support a service technician had to perform significantly less paperwork and needed less time to find the information required. Thus, the efficiency of the repair process is clearly improved. Also, according to informal feedback of test users, the usability of the application is already sufficient for practical use.

7 Conclusion and future work

We presented a new way to visualize and gather information about damage reports of a vehicle. The use of state of the art augmented reality with 3D tracking, and the use of virtual reality allow the service technician to see all the necessary information for a repair process. It shows the technician valid information about a specific damage, car parts that need to be replaced and ordered, corrosion dangers, and so on.

Thus real practical advantages compared to existing solutions have been demonstrated. In the future, the integration with the actual workshop system could be extended, for example, to be able to order parts from the Google Glass and fill in repair data on the server. Also, the overall hands-free experience could be improved by supporting additional voice commands. Then, the service technician would never have to use tapping and double-finger gestures when interacting with the application. Google Glass is still in development and receives regular updates that provide new features like wink gestures, a notification system, and sensors that can determine if the user currently looks at the screen. These will allow improving the usability of the application even further.

In conclusion, the project has shown the applicability of promising new technologies in the field of automotive maintenance and can serve as a starting point for further research concerning user acceptance and implementation technology.

References

[1] Ahmad GL. *Mobile Real-Time License Plate Recognition*. Linnaeus University - School of Computer Science, Physics and Mathematics, 2011.

[2] Azuma R, Baillot Y, Behringer R, Feiner S, Julier S, MacIntyre B. *Recent Advances in Augmented Reality. 6th ed*, pages 34–47, IEEE Comput. Graph. Appl, 2001.

[3] Azuma R. *A Survey of Augmented Reality. Teleoperators and Virtual Environments 6*, 1997.

[4] Davison AJ. *Real-time simultaneous localization and mapping with a single camera. 2nd ed*, pages 1403-1410, Ninth IEEE International Conference on Computer Vision, 2003.

[5] Gjosater T. *A taxonomy of handheld augmented reality applications*, pages 1-6, 2012 4th International Conference on Intelligent Human Computer Interaction (IHCI), 2012.

[6] Hevner AR, Chatterjee S. *Design Science Research in Information Systems*. In: *Design Research in Information Systems,* pages 9-22 Springer US, 2010.

[7] Hevner AR, March ST, Park J, Ram S. *Design science in information systems research*, pages 75-105, MIS Q. 28, 2004.

[8] Kim YD, Moon IY. *E-Training Content Delivery Networking System for Augmented Reality Car Maintenance Training Application.* In: *International Journal of Multimedia and Ubiquitous Engineering Vol. 8, No. 2.* Korea University of Technology and Education, 2013.

[9] Navab N. *Developing killer apps for industrial augmented reality. 3rd ed*, pages 16-20, IEEE Computer Graphics and Applications 24, 2004.

[10] Nishimura M, Kagawa T, Nishino H, Utsumiya K. *A Document Browser Based on a Book-Style Interface with Augmented Reality, pages 741-746* 2013 Seventh International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS), 2013.

[11] Smith R, Inc G. *An overview of the Tesseract OCR Engine*, pages 629-633, 9th IEEE Intl. Conf. on Document Analysis and Recognition (ICDAR), 2007.

[12] Lee Y, Choi J. *Tideland Animal AR: Superimposing 3D Animal Models to User Defined Targets for Augmented Reality Game*, pages 343-348 9th ed. International Journal of Multimedia and Ubiquitous Engineering, 2014.

[13] 3D Tracking based on CAD Data. . https://dev.metaio.com/sdk/tutorials/3d-tracking-based-on-cad-data/, downloaded: June 10th 2015

[14] Tutorial 10 - Object and environment tracking. https://dev.metaio.com/creator/tutorials/object-and-environment-tracking/, downloaded: June 10th 2015

[15] Metaio to Unveil First Hands-Free Car Manual on Google Glass, New 3-D Tracking Technology and optimized wearable augmented reality platform. In: Augmented Reality Blog. InsideAR,http://augmentedblog.wordpress.com/2013/09/17/metaio-insidear-google-glass-product-update/, downloaded: June 10th 2015.