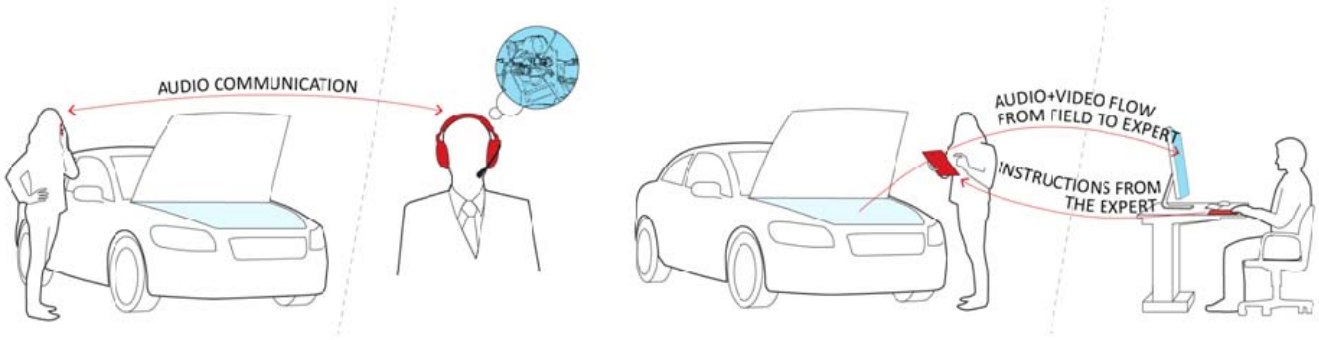
CHALMERS | UNIVERSITY OF GOTHENBURG



SEMarbeta: Mobile Sketch-Gesture-Video Remote Support for Car Drivers

Remote support for car drivers is typically offered as audio instructions only. This paper presents a mobile solution including a sketch- and gesture-video-overlay.

SICHENG CHEN &MIAO CHEN

DEPARTMENT OF APPLIED IT
CHALMERS UNIVERSITY OF TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
UNIVERSITY OF GOTHENBURG

Gothenburg, Sweden, 2014
Master thesis 2014:03

SICHENGCHEN
MIAOCHEN

Examiner: MORTEN FJELD

University of Gothenburg
Department of Computer Science and Engineering
Chalmers University of Technology
Department of APPLIEDIT
SE-41296 Göteborg
Sweden
Telephone+46 (0)31-7721000

Department of APPLIEDIT
Department of Compute Science and Engineering
Göteborg, Sweden 2014

SEMarbeta: Mobile Sketch-Gesture-Video Remote Support for Car Drivers
Remote support for car drivers is typically offered as audio instructions only. This paper presents a mobile solution including a sketch- and gesture-video-overlay.
SICHENG CHEN & MIAO CHEN
Department of APPLIED IT
Chalmers University of Technology and University of Gothenburg

Abstract

Uneven knowledge distribution is often an issue in remote support systems, and this sometimes creates the need for additional information layers extending beyond plain videoconference and shared workspaces. This paper introduces SEMarbeta, a remote support system designed for car drivers in need of help from an office-bound professional expert. We introduce a design concept and its technical implementation using low-cost hardware and augmented reality techniques. In this setup, the driver uses a portable Android tablet PC while the helper uses a stationary computer equipped with a xxx-mounted video camera capturing his gestures. Hence, oral instructions can be combined with supportive sketches and gestures added by the helper to the car-side video screenshot. To validate this concept we carried out a user-study involving two typical automotive repair tasks: checking engine oil and examining fuses. Based on these tasks and following a between-group (drivers and helpers) design, we compared voice-only with additional sketch- and gesture-overlay on video screenshot measuring objective and perceived quality of help. Results indicate that sketch- and gesture-overlay can benefit remote car support in typical breakdown situations.

Keywords: Remote support, automotive, mobile, augmented reality, handheld computer.
Index Terms: K.6.1 [Management of Computing and Information Systems]: Project and People Management—Life Cycle; K.7.m [The Computing Profession]: Miscellaneous—Ethics

Table of Contents

Table of Figure

1. Introduction

In our daily life, people always face some difficult situations, in which they cannot finish certain works by themselves, and they may turn to somebody else for helps or suggestions by making telephone calls or even video conference calls. We call this type of manners as Remote Assistance.However, we may also experience the helpless and anxiety in a remote assistance procedure. People have difficulties in describing the current situation or find the right objects via telephone, or cannot understand the helpers instruction because of lack of knowledge as an assistance requestor. Or peoplecannot describe the exact operation or objects while giving instructions as an assistance provider.

Uneven knowledge distribution is often an issue in remote assistance systems and this sometimes creates the need for additional information layers, going beyond plaintelephone calls, video conference and shared workspaces. In the described common situations in our daily life, the assistance provider in the remote assistance scenario may want to see, point out things and even show the operations to the requestor, and at the same time, the requestor may want to show the situation and having more specific instruction as well.

In this master thesis project, we made a hypothesis that added non-linguistic information, such as pictures, deictic sketches and gestures can help the requestor as well as the assistance provider to improve the experience of remote assistance. We also set a certain scenario of remote assistance, namely the car breakdown troubleshooting, and implemented a conceptual prototype application ‘SEMarbeta’ with video support technology providing sketch and gesture overlays, in order to make a comparison with the traditional telephone troubleshooting assistance. We offeredthis design concept and technical implementation building on low-cost hardware and augmented reality techniques. Hence, we offered an easily adoptable solution that should be of interest to automotive manufacturers as a built-in feature, or for end-users as a separate add-on application.

The presented remote video support technology that allows i) transferring sketch-overlaid video screenshots from the driver to the helper and ii) sending back sketch- and gesture-overlays from the remote helper to the driver. While live video is streamed from the driver-side to the helper-side, video screenshots are required for overlays. Augmented reality technologies are put at work with low-cost off-the-shelf devices to enable minor automotive breakdown cases to be fixed without requiring physical presence of support personnel.

To validate our concept we carried out a small user-study on two typical automotive repair tasks, checking engine compartments and changing fuses. We evaluated the objective and the perceived quality of help, and compared standard voice-only help vs. additional sketch-gesture-video help. Our results show positive user feedback and inform us on ways to develop the SEMarbeta system further. Our proof-of-concept prototype and the empirical results indicate that a mobile device can benefit remote support systems for car drivers in typical breakdown situations.

In the last part, we drew a discussion on the result of our user-study addressing the advantage of the introduced techniques and the flaws of our implementation of the conceptual prototype. Moreover, the scenario of usage was also discussed based on the feedback from other industries, in order to generalise the concept. Finally, we made an overview to see how this concept can be advanced by newly innovations and technologies.

2. Background

Nowadays, we all live in a more complicated world than a century before with great variety of devices around us. For industry production, one product line may consist more than hundreds of different devices and equipment, which have totally different functionality and structure. The same situation also exists in our common daily life. People may have computers with hundreds of different software, digital camera, mobile device, automobile and household appliances as well. However, this kind of technology explosion in our life is accompanied with big issues. It is not hard to find out in our life that people always complain about that they cannot or do not know how to maneuver their devices. This kind of situation also exist under the industrial context that the user or operator of the equipment always have insufficient knowledge and understanding to fix the equipment when it breakdowns.

We can provide many similar scenarios that people face this kind of knowledge barriers. For example, one may just buy a new computer but cannot access to his network connection. He tried to use diagnosis tools but it only shows some error code, which cannot be understood. He called the customer service of his ISP provider, and the technician answers his question with some instruction. During the assistance process, he still cannot fix the problem because of too many professional vocabularies, such as static IP, dynamic IP, subnet mask etc., or and he cannot find the application and properties as the expert asked. Finally, he must call for a door-to-door service in order to fix a simple network problem.

Actually, the situation of solving problems related to computer context becomes better with the help of tele-assist tools. However, this kind of remote access solution cannot solve the big part left on devices without direct network connection or operating systems, such as our household appliances or automobiles. Is there any solution that can help non-professional people to fix their problems, and replace the traditional phone-call assistance?

When we move our eyes over the industrial and medical context, this kind of issue become much worse. For instance, if the device is imported from other country, or a patient has to take an operation but the expert in this area is far away, the fact of requiring the expert to be on-site may bring enormous extra cost. Moreover, some of the collaboration nowadays can be taken remotely with videoconference, but for tasks that require complicated operations, such as design, surgical operation and repairing, the face-to-face video can only bring limited help.

Inspired by this issue, we aimed at design and implement a solution that can release the difficulties in some degree. Because of the time and workload limitation of master thesis work, we chose automotive repair as our use case and background context. We also collaborated with industry professionals in the automotive, customer information and design area, in order to build more realistic requirements on the solution, rather than based on imagined curriculums.

3. Related Work and Theory

According to some similar scenarios research as we mention in background, which we deeply believed on only voice talk is not satisfied most situation in reality. Even if a face to face video meeting, it cannot change anything. Many misunderstandings and conflicts are due to poor communication caused. Sometimes, even two sides have same educational background; they also cannot very good to understand another side due to unclearly voice talk. So we can imagine, if two sides have different educational background or cognitive levels, the result of communication could be worse. As the result we try to find out a good solution that can improve or solve above issues. However, there are many different bad situations, which can be researched in this world, we cannot research them all. In addition, we signed a contract with an automotive design company, thus we can only research the problem in automotive field. Besides, we will also looking for some useful related works which in order to contribute suitable theory foundation for this thesis work.

3.1.Related applications

Since the research was done in collaboration with an automotive design company, we adopted the design method of participatory design. Senior engineers from the company presented requirements and provided professional suggestions based on their long experience on automotive design and information presentation. Hence, we developed the idea of a system that leverages mobile devices to offer car drivers remote technical support. We researched existing remote support services in the automotive field, as well as automotive support applications provided in IOS App stores and in the Android market (See Figure 1). We found that the envisioned remote support system for drivers could have a huge potential since the current solutions all depend on voice-only support. We also found that customers in the automotive industry tend to prefer multipurpose solutions where mobile devices not only offer remote support but also provide remote control for other car functionality such as an infotainment system.

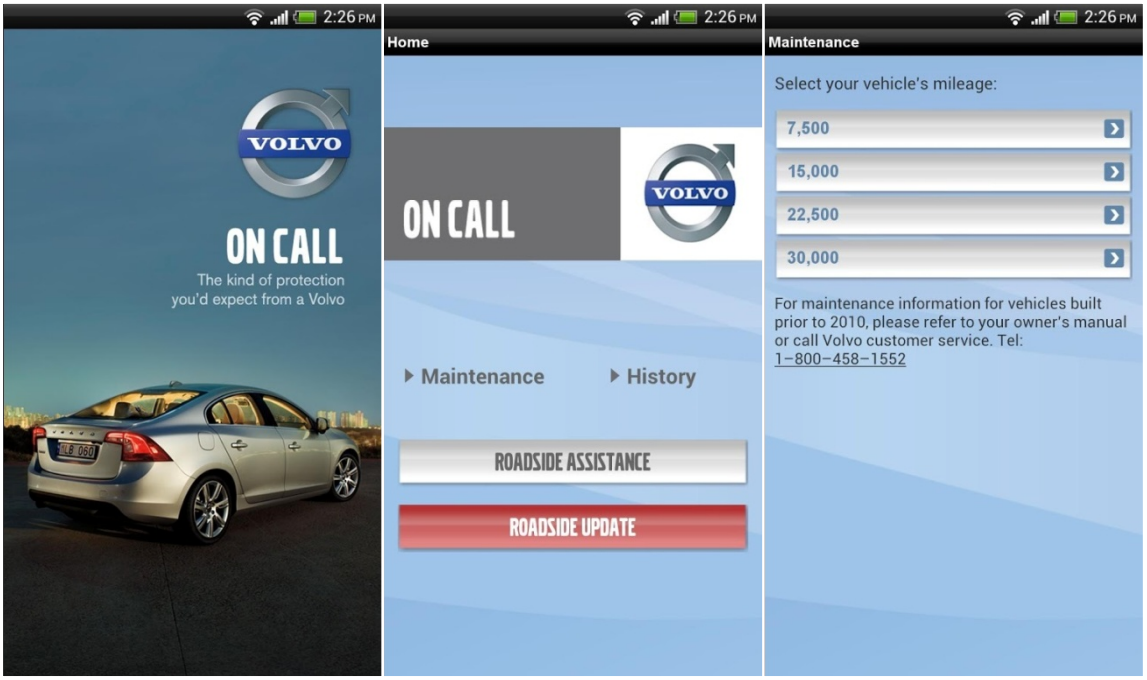


Figure 1 Screen shots of Volvo on Call application on Android

3.2.Related researches

Besides taking participatory design with the professionals from our industrial partner, we also took a broad view into the existing technologies in the collaboration & assistance area.

The mobile collaboration research area focuses on utilizing mobile devices to develop remote collaboration and training systems. Papadopoulos emphasized that the group awareness of watching the others' activities and by coordinating them already satisfies the need of collaboration [5]. Moreover, the research done by K. O'Hara et al. shows that a video image can reinforce the affective experience in communication between geographically separated collaborators [6]. Also, the research done by V. Herskovic et al. also reveals the different modes in mobile collaborative systems [7]. In the distant learning research field, mobile collaborative tools also play important roles. Daniel Spikol et al. developed devices that give access to educational resources and allow collaborative learning outside the classroom [8].

Previous research in the remote support system field always aimed to design and develop systems or services for particular areas and users. ReMoTe [1], for example, is a remote support system for instructing miners working underground. The helper side of that system can get a visual understanding of the working situation by viewing the live-video captured by the worker's head-mounted camera. Meanwhile it also captures the helper's instructions from his display and sends it back to the worker's side.

Augmented reality technology has delved into the automotive industry over the course of many years, especially in improving the efficiency of automotive assembly work. ARVIKA [4] was a large research project on adapting augmented reality technologies to the automotive field. The system uses head-mounted displays together with marker-based tracking in order to support service and training tasks during car inspection. Research done by Anastassova and Burkhardt [9] also reveals the current training structure and provides guidelines for the future AR implementation in automotive assembly training. Besides the automotive assembly field, AR can also be a meaningful alternative in the design industry, helping designers to express their innovative ideas and overcome technical difficulties, which is revealed in the research of Ran et al. [10].

In the remote collaboration field, the mixed reality technology also contributes to enhancing group awareness between geographically separated collaborators. VideoArms [2] and CollaBoard [3] present solutions of live-video overlay on shared workspaces, in order to create a virtual side-by-side impression among collaborators. While VideoArms shows collaborators hands and arms, CollaBoard transmits the image of the collaborator's upper body.

In the beginning, it was difficult for us to build a general view of these different minor subcategories in the collaboration & assistance area, because all of them sharing the same concept of building a way of communication or knowledge sharing. However, after digging into the details, we found out some essential differences within these subareas.

We took three properties of these technologies, namely Mobility, Flexibility and Collaboration. With these three properties as standards, we can easily build a general view to understand the connection and differences within them. It is obvious to distinguish low mobility and high mobility, which means with higher mobility, both sides of the collaboration or assistant works can move freely without being stationary. For the concept of flexibility, it presents whether the information or knowledge that available for the users shall be prepared or predefined. It also means the ability of the technology to solve emergent issues depending on the current situation, rather than following existing process. The degree of collaboration can represent the knowledge sharing modes. For example, full collaboration means equivalent level of knowledge sharing. Both sides of the communication provide their personal knowledge and understanding and get the other's input in return as well. In contrary, for assistance, the side being helped cannot provide its knowledge or understanding on the issue, with only the helper providing one-way input, so it has low degree of collaboration.

3.3.Theory input from related researches

The concept of the ReMoTe system is similar to our research objective, in which the helper side can provide additional information to the worker side besides just simple linguistic instruction. However, the concept of designing a system and equipment for professional users is not the subject of our research. The head-mounted display and camera are exclusive equipment for miners, which are not suitable for everyday use. Further, the head-mounted device is not a see-through device and thus does not provide an augmented reality overlay of the helper's hand image combined with the live-video image on the helper's screen. However, Google's glass project [11] might be a future device from which our envisioned system could significantly benefit.

Other augmented reality solutions in the automotive field, like ARVIKA [4], depend on pre-defined tags to track position and to provide 3D animations. This kind of solution is not suitable for the remote maintenance situation, because real-time diagnosis and instruction are needed in a daily application and in an environment that does not have any visual markers.

Our system design was also inspired by the CollaBoard research project. Within this system, the full upper body of the collaborator is displayed on the other side. Thus, all information like postures or deictic gestures is in context with the underlying content of the whiteboard. Although we do not need postures, transferring of deictic gestures is crucial for our system, since it is the most natural explanatory gesture. However, since we want to use mobile devices, an important design aspect is the size of the screen. This should be large enough for the driver to unequivocally recognize the helper's gestures in relation to the underlying image, while still having a handheld portable device.

4.Methodology

In the forming procedure of this master thesis project, we found out an interesting issue that many projects done by students cannot reach a higher applied level and such topics stayed at the same situation without updates for many years. For that concerning, we decided to expand our project from the laboratory range to the industry, which also means reinforcing the collaboration between the students and professionals in industry, taking professional advice to make the project more sustainable. From the aspect of sustainability, the corporation with industry partnership does not only assure the continuousness of the project topic, but also generate the possibility of taking new ideas into the future industry production.

The design process of this master thesis project covered multiple methods functioning on different stages in the project. In this master thesis study, we decided to adopt the Stanford Design Process [17], in which a design project was decided into six stages, namely **Understand, Observe, Define, Ideate, Prototype and Test**.

4.1. Understand

In the design process model suggested by Stanford University, the goal of this stage is to get experience addressed on the topic from the experts and to conduct the research. For us, the challenge is not only gathering experience from experts, but also selling the topic to a proper industrial partner and its experts.

4.1.1. **Storytelling**

In order to find an industrial partner who is interested in the topic of remote assistance system, we have to let those experts within the area to get the empathy of realizing the necessity of remote assistant system or even an augmented reality solution of it.

We decided to use the method named Storytelling. In the article “Storytelling Group – a co-design method for service design”, Anu Kankainen et al. have emphasized the method Storytelling, which means users telling real-life stories about their experiences, is really helped in the defining of a point of view, the desires and needs. Besides, this method can also reflect our original attitudes telling why we chose this topic of remote assistance as the research topic of our master thesis.

Since real life stories are most persuasive, and both of us as researchers have the real life experience of desiring a remote assistance system. We just told our target industrial partners two real stories of us.

Story 1: Sicheng’s father works as an electric engineer in China. In Sicheng’s childhood, he has strong impression that his father always travelled to the customers for field works to fix welding machines. However, in those costing field works trips, many of the problems actually are caused by mis-manipulation or are not that hard to be fixed by the operators themselves. Thus, his father always mentioned the desire of having a remote assistance system to help him to do a more precise diagnosis in distance, or even to instruct operators to fix minor problems.

Story 2: Miao came from a doctor's family, and his father is a local expert with well reputation. In China, the medical resource is not that adequate, which means as an expert, Miao's father has to do many consultations in diagnosis and surgeries. As his father told, he always experienced the situation, which is hard to describe a certain position or an operation in the consultation procedure. Miao's father expected a new medical consultation system helping him to point out objects directly and to show the right operation to the advocate knife doctor.

4.1.2. **Participatory design (Stakeholders)**

It took several months in looking for an interested industrial partner in Gothenburg, and finally Semcon decided to build the collaboration with us on doing this research with holding the intellectual property outcomes of this research.

Semcon is an international technology company based in Gothenburg, Sweden, and it is active in the engineering services, design and product information. Most of the biggest customers are automotive manufactures and component suppliers.

Based on these facts, we had to redirect the topic a bit to make it more realizable to our industrial partner. We adopted the design method of participatory design, which means the approach to design attempting to include and respect the current situations and needs of stakeholders (employees, partners, customers and users). Therefore, we ran several workshops in this stage with the business responsible persons and engineers at Semcon. They showed great interesting in this topic and wondered how the concept can provide helps in the automotive industry.

Two use scenarios were raised by the workshops, a road car breakdown scenario and a remote training in the car manufacturing factory scenario. Due to the time plan and cost limitations, we chose the car breakdown scenario to expand this master thesis research.

4.2. Observe

This stage of Observe stands for a design phase in the process, in which researchers should watch how people or target users behave in the physical spaces. Researchers may also get a better understanding from this phase and develop the sense of empathy.

In this stage, we did the observation in three different ways. The first round of observation was done by watching illustration videos, and the videos we gathered from on-line sources showed different sort of problems caused breakdowns and the behaviors made by drivers in those videos. In those videos, we got one distinct impression that even ever manufacturers printed and sent out driver's manuals with the car, however rear of those drivers showed that they have read and understand the manuals before breakdown happened. Another characteristic shown was that people were more inclined to ask for help from others, such as first-aid service, relatives and other road users rather than reading manuals. Finally, people in the videos who had made telephone calls for help always had trouble to describe the cause of breakdown and to follow the provided instruction because of lack of knowledge.

The second round of our observation was done by interviews of people who have used telephone first-aid services. The interviewees reflected similar result with what we have observed from the first round, and they said it was still quite hard to fix the breakdown cars by themselves even the assistance understood what the errors were, because the drivers could not fully understand and perform the told instructions.

The third round was to experience a car breakdown on the real road and tried to make a call for help. In this practice, we found out it was really hard for the both side (drivers and assistance) to get a deeper understanding the breakdown scenario by describing via telephone calls. This trouble was more severe if the assistance had no experience on the type of the car or the certain model variant. In the repair, we also experienced the trouble in providing instructions. The most common situation was, the assistance asked the driver to look for a certain object, and then both of them had to describe the color, size, shape or even materials to confirm they were meaning the same object. After that, many operations related to automotive were not that common in our daily life, and some of them might need to use tools. The assistance must clarify an operation by using unprofessional vocabularies, and address the properties of this operation, such as needed force, orientation, speed and some skills. Besides, the assistance had to always ask and confirm the driver performed the instruction right repeatedly.

4.3. Define

From the Design and Observation round, we can generate a set of requirements for the target system.

- The hardware setup shall be possible to be implemented with on-the-shelf components.
- The driver is willing to point out and explain the troublesome question in the request.
- The expert is willing to look at the problematic scene.
- The expert is willing to show the operations and tools of solution.

We used the Use-scenario method (See Figure 2) to visualize and polish the definition. For this, we produced a short video showcasing a typical use of our envisioned system. In the video (See Figure 3), a driver faces an engine breakdown. He picks up his mobile device and starts an app provided by the car manufacturer. After having connected with the helper side, the expert provides instructions with sketching and gesturing. Based on this demonstration it became easier for the senior engineers to understand the overall concept of our system, and thus be able to raise more specifications, such as reducing the use-cost of the helper side, defining the handheld device that can be a standard automobile feature in the future, and so on.

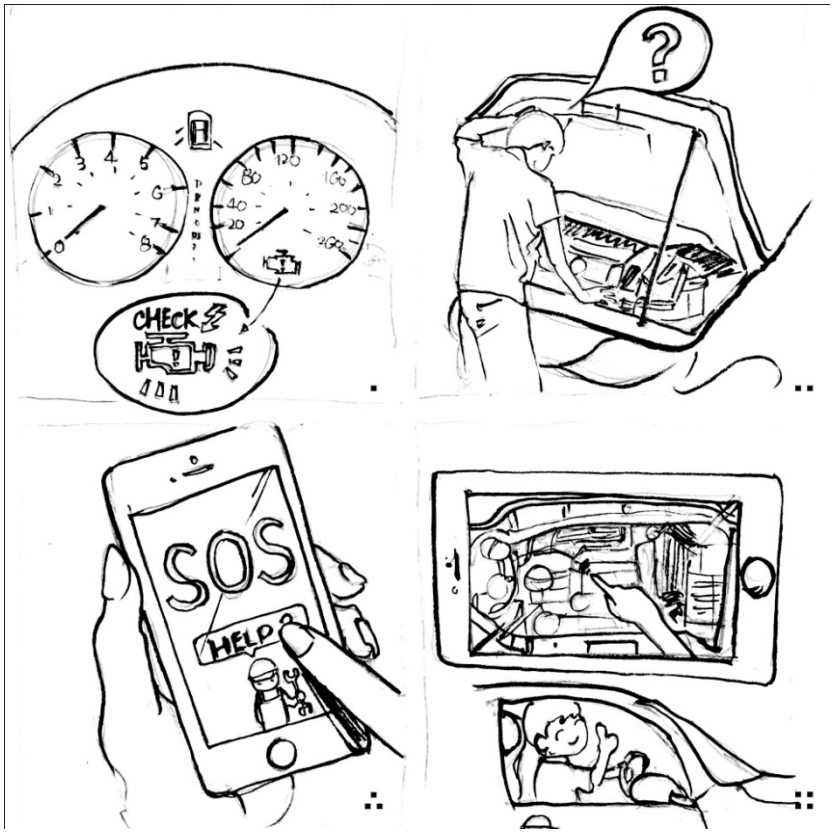


Figure 2 Storyboard of the user scenario

Figure 3 Screen shot of our scenario video.

4.4. Ideate & Prototype

See chapter 5, 6, 7;

4.5. Test

See chapter 8, 9;

5.System Concept and Realization

The final system concept is generated through a long procedure of participatory design with idea iterations and literature review. Two more technical steps followed: hardware architecture and strategies for vision-based gesture capturing.

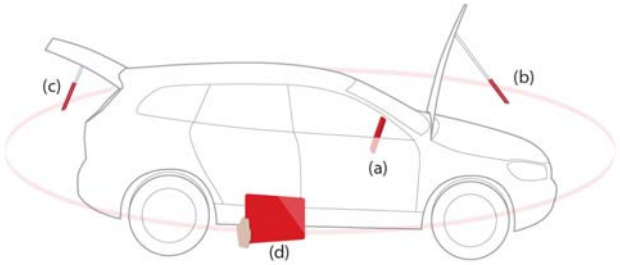


Figure 4 Different usages and setups of the mobile device

The participatory design workshop indicated that mobile devices would have to be multipurpose. For example (See Figure 4), the same device should offer infotainment functions (a), support engine (b) or luggage (c) compartment instructions, and potentially instruct drivers how to change tires (d).

Figure 5 A standard information system facility of the car

There is a trend of having infotainment system in cars as a standard facility in the following year (See Figure 5). With infotainment system in the car, the driver can access to the control of the climate system as well as the media center, but also can get geographic information and other functions, such as GPS, weather, news, personal contact and so on. Infotainment system is becoming a more compact and complex platform that is a good stage for our system to stay in, in order to help the drive to face emergent events, namely breakdowns and accidents.

5.1.Hardware Architecture

Our system consists of a handheld device (tablet) and a stationary computer (See Figure 6). Both the tablet and the stationary computer offer sketching and audio communication. The driver side can transmit live video streams to the helper side in order to describe the problem (e.g. check oil, locate fuse). The helper receives live video streams from the remote situation (e.g. the car engine or fuse board) and can give instructions on how to check oil or fuses, either by outlining directly on his screen or by using gestures that are captured by a camera. His sketched outlines and gestured instructions are directly overlaid on the live video stream and can also be seen by the driver. By sketching with another color, the driver can also outline in order to clarify problems.

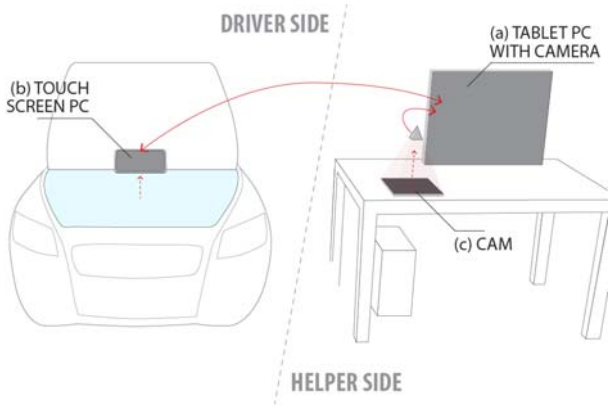


Figure 6 Hardware architecture.

An interesting issue occurred in the procedure of deciding the mobile device for the driver side. Having smart mobile telephone on hand is easy for a driver to carry it and working around the car in the repairing. However, a screen around 4 inches is not big enough for human eyes to recognize the live instructions (sketching and gesture), even it has a really high resolution as retina display. In contrary, a tablet PC can have bigger screen that can contain the gesture image in a good fit. However, the heavier weight and bigger physical size may bring difficulties for the driver to hold with only one hand while he is working on the repair works with the other hand. Finally, with the consideration of merging the system into future infotainment system as the target and good presentation for having mixed reality overlay, we chose the tablet PC as the device for the driver side.

5.2.Alternative Strategies for Gesture Capturing

Gesture capturing in front of a highly dynamic background such as a live video is a delicate task. While VideoArms used color segmentation algorithms to capture the deictic hand-gestures in front of a screen, CollaBoard used a linearly polarizing filter in front of the camera and thus benefits from the fact that an LC-screen already emits linearly polarized light (See Figure 7). However, both methods have their own shortcomings. While color segmentation only works well if no skin-like colors exist on the screen, the solution with polarized light cannot detect dark objects that do not significantly differ from the dark-gray of the captured image of the screen.

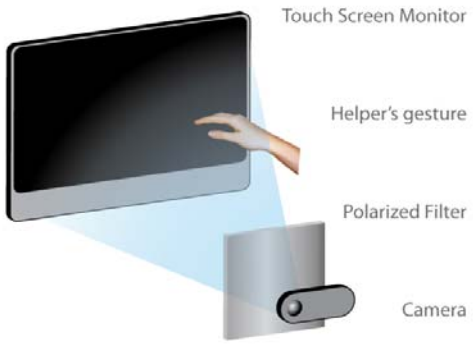


Figure 7 One example of gesture capturing solution

Although our system could be adapted to both of these segmentation strategies, we adopted another solution to capture the helper's gestures. We did not use any polarizing filter or color segmentations. Instead, we hang the camera aside the stationary computer facing downwards and placed a black mat (16"x12") below the camera. Thus, the helper can put his hand between the camera and the mat for his gestures to be captured. Here, we take benefit from the fact that we are used to this spatially distributed interaction. We can control mouse pointers or perform pointing gestures even if we see the results indirectly on a separate screen. However, there are some limitations to this method, as we will discuss in later sections.

6.Hardware Implementation

This section will describe the selection of hardware depending on the requirements on our system. The selection of hardware was guided by the principle of using only inexpensive hardware. Furthermore, we took into account that the driver's mobile unit should be light and also usable for other tasks. Unlike VideoArms or CollaBoard we propose an asymmetric setup while still using many of the CollaBoard features. Two sides are involved in our system (helper side and driver side) as well, but each side works in different situations (See Figure 6).Since the two sides in our system design may have unparalleled knowledge distribution, helper side may afford more workload of input than the driver side. Thus, there are two different system setups in our system, i.e., a helper side and a driver side, which are presented next.

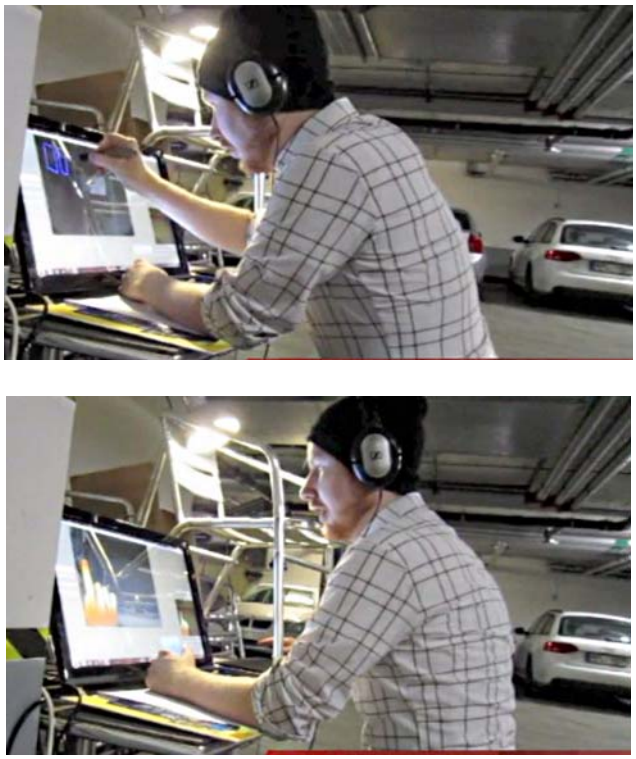


Figure 8 Implementation of the helper side: Sketching (top) and gesturing (bottom).

6.1.Helper Side

On the helper side (See Figure 8), only the image from the driver side has to be displayed, and touch input has to be detected. Thus, standard touch screens are sufficient. In the prototype, a 22" touch screen is used.
Since we use a touch screen, no further input devices, such as mouse or keyboard are needed, and the helper can easily use a pen or finger to interact with the software. However, since gestures should also be captured and transferred, an additional input capability is required.

Like CollaBoard and VideoArms, the SEMarbeta system also captures the helper's gestures (deictic instructions). Therefore, a camera is needed in order to capture these as well. Considering the cost and quality of cameras, we selected a high-resolution webcam with an auto-focus function for our prototype (Logitech QuickCam Vision

Pro 9000). For our first-prototype, there was no need to apply polarizing filters to eliminate the background image of an LC-screen, since a different setup for gesture capturing was implemented. This is discussed in the software design section below.

No specific setup for the audio channel was required. The default microphone in the camera is used for audio input. For clearer audio quality, headphones are connected to the audio output.

The helper side application is running under Windows 7 OS. Since the application runs an image processing function as well as a video transmission, a powerful CPU (Intel i5 CPU) is required. The computer is connected to the LAN.

6.2.Driver Side

Due to mobility reasons, the driver must be able to hold the device easily and conveniently. Moreover, the device has to run an image processing in order to guarantee a smooth video transfer. In our prototype, a Samsung Galaxy Tab 10.1 [12] was chosen. It provides multiple network connections (Wi-Fi and 3G), so that the driver can connect at any time as long as a network is available. The device also has two cameras: one on the front and on the back; the back one was used to capture breakdown situations.

In this research, since the gesture images provided by helper side is emphasized, the driver side device is requested to be big enough maintaining gestures that are clear enough to express operations. For that reason, we chose a tablet PC rather than a smart mobile phone. Another reason is we aimed at introducing such system to the automotive industry, so the examination of our remote assistance system on portable infotainment system is also an interesting point.

7.Software Design and Implementation

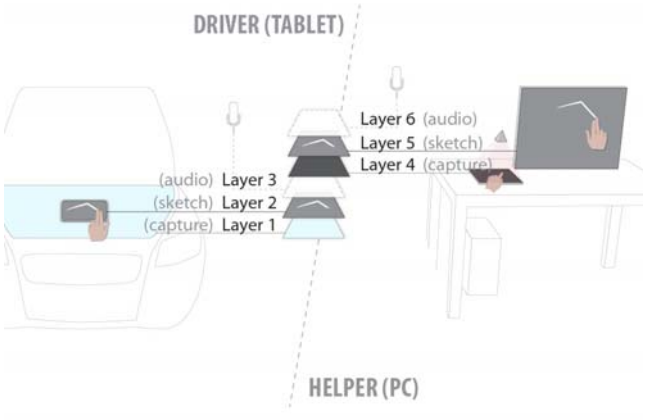


Figure 9 Functional Layers in the software

To our knowledge, Microsoft does not support the ConferenceXP [13] remote presentation software anymore (as it was used in the CollaBoard system). Therefore, new software was developed for our prototype, which can be used for the remote support system. This software provides three different information layers at each side (audio, sketching, and image capturing). In our software architecture, the driver will use Layers 1-3, while the helper will use Layers 4-6. (See Figure 9)

Since our prototype software is a remote support system, which contains the helper side running on stationary computer and the driver side running on mobile device. For that reason, it requires the system can run on different operating systems. Because of an industrial thinking, we chose Windows OS as the running environment for the helper side. The Windows OS has been widely adopted in a business context and supported by budget devices. Actually in the earlier stage of the designing, we thought about taking the Linux or MacOS as the running environment, since there are more open-source remote communication libraries available. However, the Linux system is not quite suitable for common users because of more complex operation and too many different versions of OEM. The MacOS is neglected because of its incompatibility and higher price of the devices. For the driver side, the device has limited our options at first. As described above, we chose Android tablet PC as the mobile device for our system, then the running environment must be android 3.0 or higher. Another reason of choosing Android OS is also from the industrial thinking. In the automotive industry, Android system has been widely used by automotive manufactures for their infotainment system presented by center stack displays. Moreover, we tried to avoid extra budget for the software implementation, but the iOS asks developer to pay for the authority if he wants to test the software on actual device.

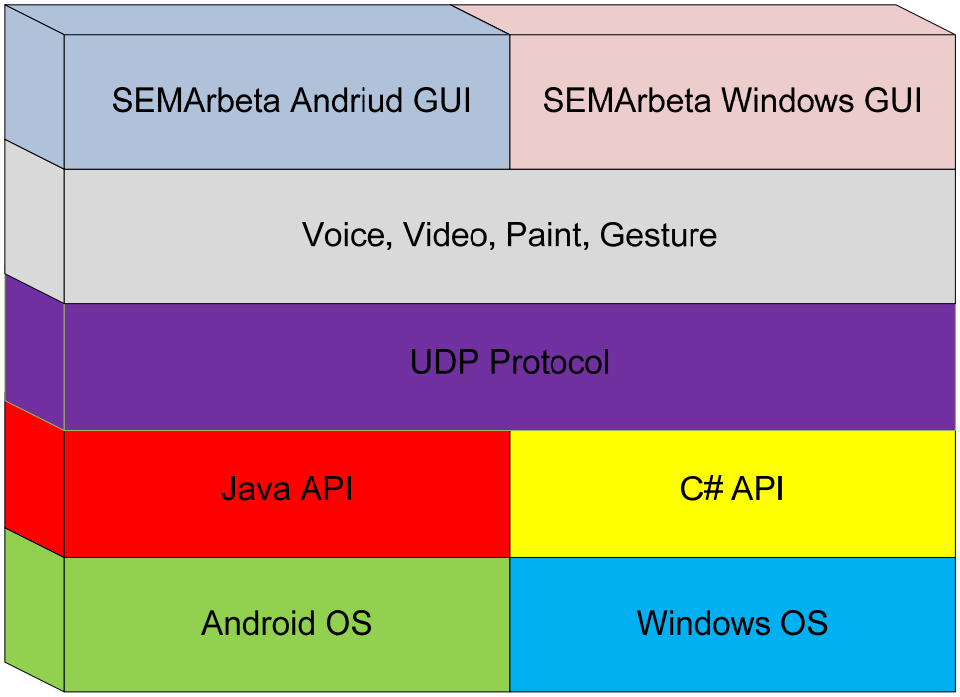


Figure 10 Software architecture

As the figure shows (See Figure 10), we adopted Android API as the developing environment, which is mainly the same with standard Java API. For the helper side development, we chose C# for programing for an easy and fast GUI implementation; even we have multiple choices available on the Windows OS.

Next, we describe implementation and functionality of our software running on both helper and driver sides. We also present and reflect upon how the user interaction was realized.

7.1.Audio and Video Connection

In order to realize a smooth video and audio transfer, the UDP [14] protocol is adopted for the transmission. A driver with a technical problem in the car can directly start a VoIP [15] call to the helper side, while the helper can decide whether to accept the call or not. When accepted, audio is first transmitted to the other side. After the helper and the driver established an initial communication, they can activate a live-video stream in case the helper thinks the problem is too difficult to be explained by voice only, or if the driver considers the problem too difficult to describe. In this case, the Samsung Galaxy Tab transmits the working scene to the helper side.

Since in this master thesis project we only implement a functional prototype, we haven't paid too much attention to the network communication optimization. For that reason, we adopted UDP protocol for its no latency sending, even if it may lose some packages in the transmission. TCP protocol performs better on the package lost problem but it may lead to higher latency in the video communication if the network environment is not stable.

During our implementation, we found it would lose packages if the data for each frame in the video streaming were big. Because of this, we spilt each frame into

multiple sections of data then transmit them in the software. The first step is to define a proper length of data section, based on the speed of transmission and proper buffering size for the device. After this, the software calculates the number of sections for this frame and then sends this information combined with the data to the other side. The advantage is the receiving effect of the other side will not be severely influenced by the network condition even though some data are lost in the transmission. For example, the driver side sends a frame in the video streaming to the helper side. This frame is split into 6 sections but the other side only receives 5 of them. Thus, this frame will be shown on the screen of the other side with 1/6 of pixels are grey or unchanged. However, if we send the whole frame without fragmentation, the other side cannot show this frame when some data of this frame is lost. The helper side will get blank only on its screen.

7.2.Screen-shot functionality

The reason for having the screen-shot functionality is obvious. As we see it, the driver has to hold the device in one hand while doing repairing operations with the other hand. This would result in a very unsteady video image. As a result of the user test, we found that it is very difficult for both sides to point at the same thing or to outline objects in a live video. Even the slightest movement of the device would disturb the analysis of the problem by the helper and thus hinder the discussion. Thus, we designed and implemented a screen-shot functionality in the driver side application and the Android tablet can temporally freeze the screen, so that the helper and driver can discuss the issues with a steady image.

7.3.Sketch overlay

The sketch overlay is one of the essential functions in our system. Within the communication between the helper and the driver, it is still very difficult for the helper to explain problematic issues by audio only, even if the helper recognizes the problem. This is mainly because of an uneven knowledge distribution between the helper and the driver. Since sketches can much easier explain certain issues, we realized the sketch overlay. It has basic sketching tools for both the helper and the driver and allows outlining the issues directly, which will then be transferred to the other side (See Figure 11).

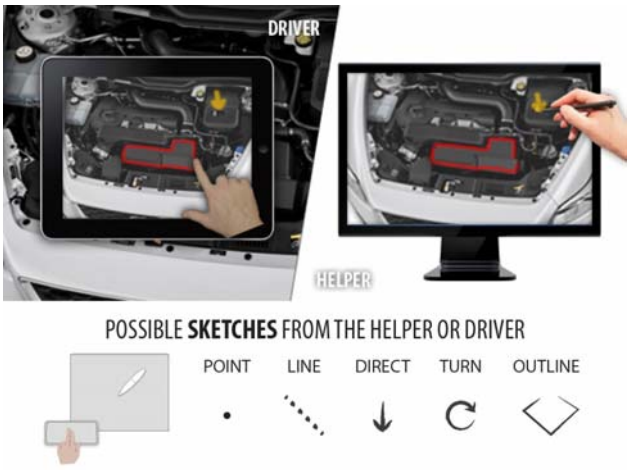


Figure 11 Implementation of the helper side: Sketching (top) and gesturing (bottom).

7.4.Gesture overlay

When a troubleshooting strategy is hard to explain, using hand gestures may help in clarifying the situation. The real gesture image or animation can provide additional information. It does not only assist the helper to clarify some specific operations that are hard to explain with speaking, but also make the instruction easier for the one being helped to understand and carry out the operation. However, deictic (e.g. “this handle” or “that fuse”) gestures are only relevant when shown in relation to the problem, that is, to the underlying image (See Figure 12).

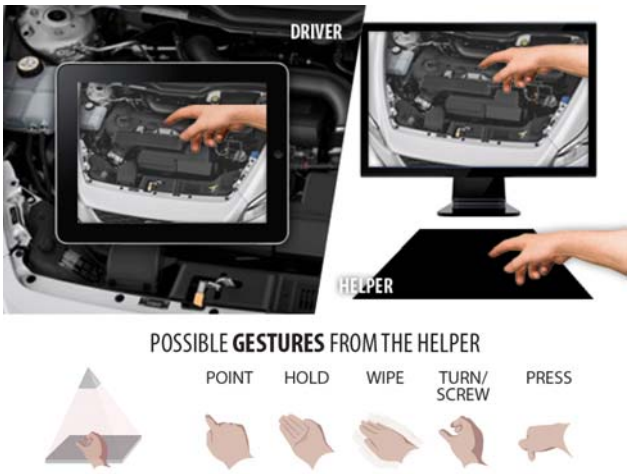


Figure 12 Gesture overlay function implementation (top) and possible gestures that can be used by the helper (bottom).

Because of the time limit, budget and current technology limitations, the gesture function is only available on the helper side, which means on the helper side can capture and provide gesture to the driver side in our prototype. However, since the system is facing the remote assistance system area, in which the knowledge distribution is not equivalent between both sides, the driver side does not need to provide the same amount of input to the other side, and the effect of having gesture on this side is decreased.

To capture a hand gesture but not the local background, it must be segmented from the background. We chose an image processing function where the software captures an image of the hand in front of a unique black background. Then, a gray-scale function transforms the whole image into gray-scale image (See Figure 13).

Figure 13 Transformation to grayscale image

After that, a mask function transforms the gray-scale image into a mask image, depending on a specific threshold value. Pixel gray-scale values below the threshold are set to 0 (black); values above are set to 255 (white). Since the background of the gesture is a black mat, the gesturing hand is white while the background is black within the mask image. In the software implementation on the helper side, the threshold value can be tuned in real-time by the helper to get a better mask depending on the current work illumination condition (See Figure 14).

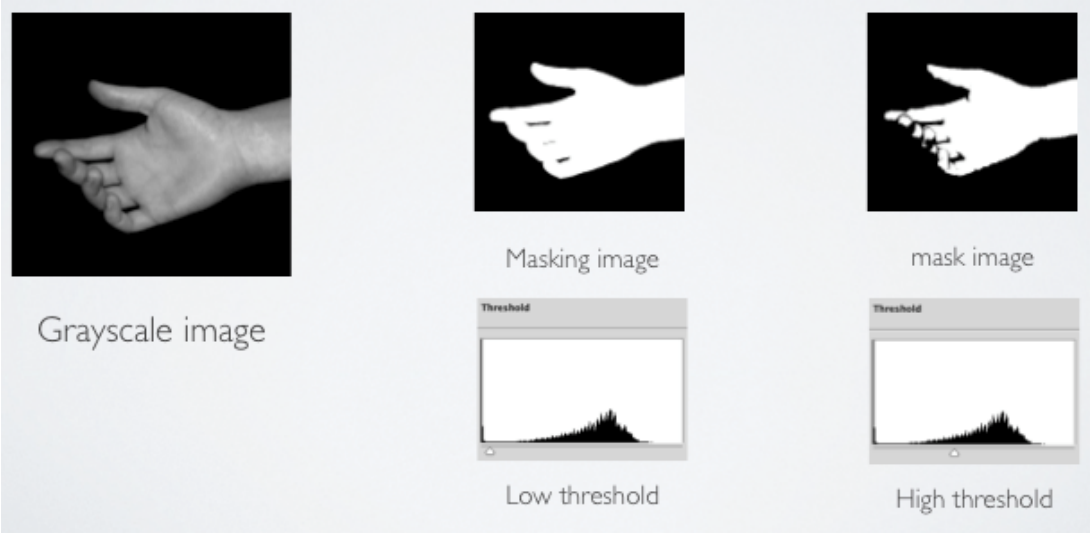


Figure 14 Threshold adaption

In a next step, another function named ‘processing’ compares the original image with the mask image. If the color of pixel is black in the mask image, then the color of that pixel in the original image shall become transparent as the concept. While in the real practice, because of the information of having alpha layer for images may bring in

extra data cost in the transmission, it is better to replace the transparent pixels with extraordinary colors, such as light green, which is quite often used for digital photography for its rare existence on human body. This replacement helps the system to avoid unnecessary cost in the data transmission and keep the video streaming being fluent (See Figure 15).

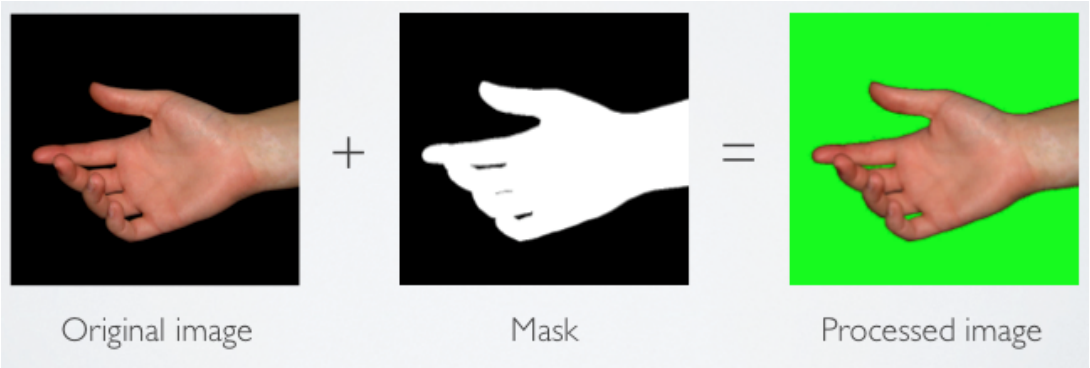


Figure 15 Apply the mask image on original image.

The final result of this segmentation pipeline is an image of the hand in full color against a transparent background, which will then be overlaid atop the still image from the driver side. In the implementation of the driver side, the received image will be processed in order to eliminate the light green pixels and overlaid on the video layer. One flaw of the current practice is the margin of the gesture image is green-like color, caused by the compression of image on the helper side before transmission. The pixels on the margin are not precise light green after being compressed and anti-aliasing (See Figure 16).

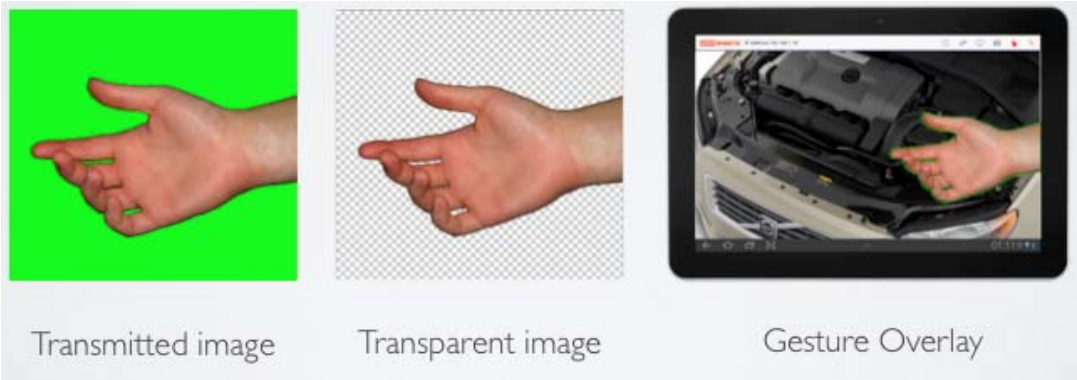


Figure 16 Overlay the transmitted image on captured scene.

7.5.Driver side GUI design

Because the driver side of this remote assistance system is used by non-expert users, who are not use the software daily and have limited time to get professional training. We decided to put more efforts on designing user-friendly graphic user interface on this side, and leave the helper side software with a prototypical graphic user interface. This decision is also led by the limitation on time of master thesis study

7.5.1. Case study

Since SEMarbeta is remote video/audio assistance system acting on different platforms, we chose two famous and common use videoconference application on multi-platforms, Skype (See Figure 17) and Google Hangout (See Figure 18).

Figure 17 Skype on Android

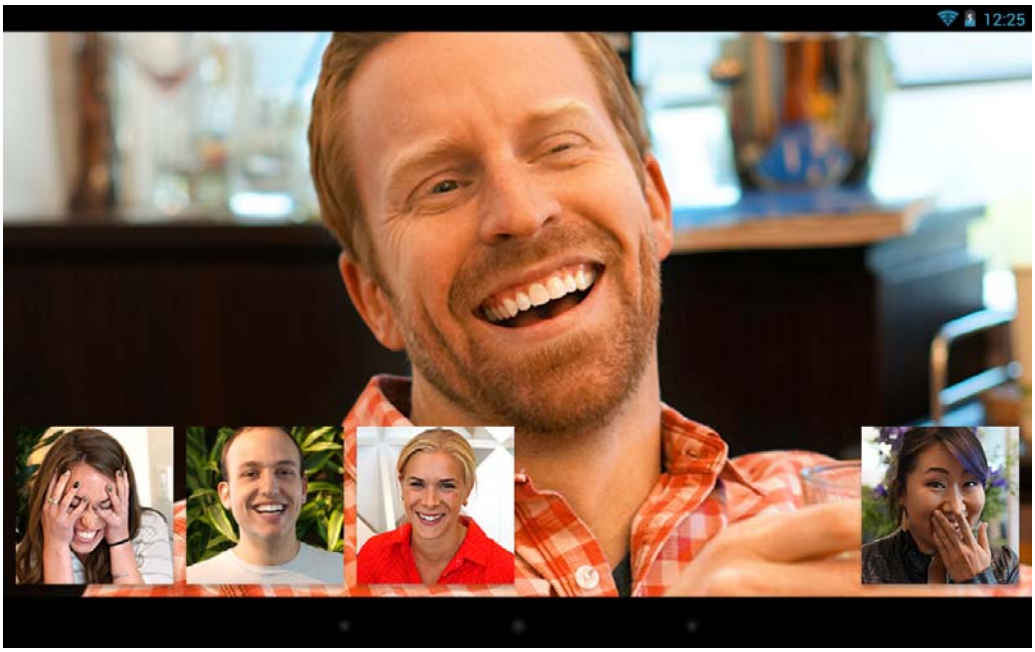


Figure 18 Google hangout on Android

It's not difficult to find out their similarities in-between from their screen-shots. As video chat or conference applications, the area for video presentation occupies most of the screen. The header of the contact person is placed at the bottom of the screen. For Skype, the header can be moved by the user to the other corners on the screen.

Since the Google hangout is available for multi-chat, more than one headeris at the bottom. One difference that we can see from the screen shots is control. In Skype, it places four buttons to control the video chat, such as turn on/off video source, mute, text and switch off the conversation. On the other hand, the Google hangout conceals the control dock while the conversation is on, and the user must touch the screen to summon the control dock if he/she may want to quit the conversation or do other settings.

However, those inspirations from the existing videoconference software are not capable to cover all the function requirements of SEMarbeta. Thus, we started to look for the videoconference software allowing painting or notes overlay.The Google hangout on web happened to be a good example from this aspect (See Figure 19).

Figure 19 Google hangout on web

The web on-line version of Google hangout enables add-on overlay showing gadgets and pictures. Tools are docked on the lefts side of the window and folded automatically. However, when we compared the layout design between the web version and the Android version of Google hangout, we found out the switches, such as the mute, video on/off and settings buttons are located on the top of the window in the web window, instead of showing them over the video image.

7.5.2. GUI design

Based on the case studies described, we created three rules for our graphical user interface design, namely:

1. Optimize the GUI design for tablet computers.
2. Save screen space for the video chat content;
3. One-click to mainly functions, and put infrequently used functions into submenus;

7.5.2.1. Layout

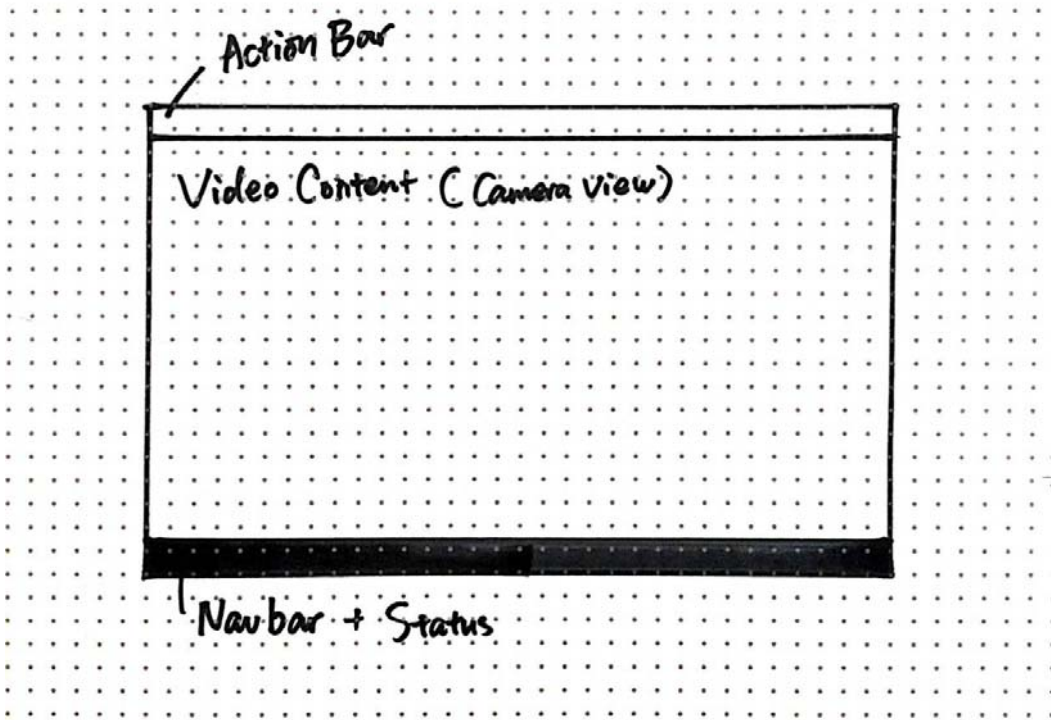


Figure 20 Wireframe of the application layout.

By following the GUI design guides provided by Google, the layout of our driver side app may contain three big sections: the action bar, the content panel and the default Android OS navbar (See Figure 20).
Since the navbar section is provided and fixed by the operating system by default, and changes on that part is not recommended, most of the GUI design happened on the action bar section and content panel section. Based on the second rule made from previous studies, we decided to keep the content section clear with no acting buttons or information windows on it. Because the user also need to make sketch and notes overlays on the video contents as said, and such actions may perform by 'point' and 'touch' behaviors, putting buttons overlaid the video content must create collisions in the operation. Therefore, action buttons or any further actions besides sketching shall be placed on the action bar.

As suggested by the developers' guides of Android, action buttons or action overflow is pinned to the right side. Besides, the numbers of action buttons are also suggested by Android, showing 5 icons on a 10'' tablet would function well. *(reference* http://developer.android.com/design/patterns/actionbar.html*)*

We have run some small tests with testers in both gender with different sizes of hands, and we gathered their feedback on holding a 10'' tablet horizontally with two hands. Imagine the tester is right-handed, his/her 'comfort zone' of making touch actions may look as the figure shows. This result from holding tests is in accordance with the

Android's guidelines suggesting that action buttons shall be placed to the right side.

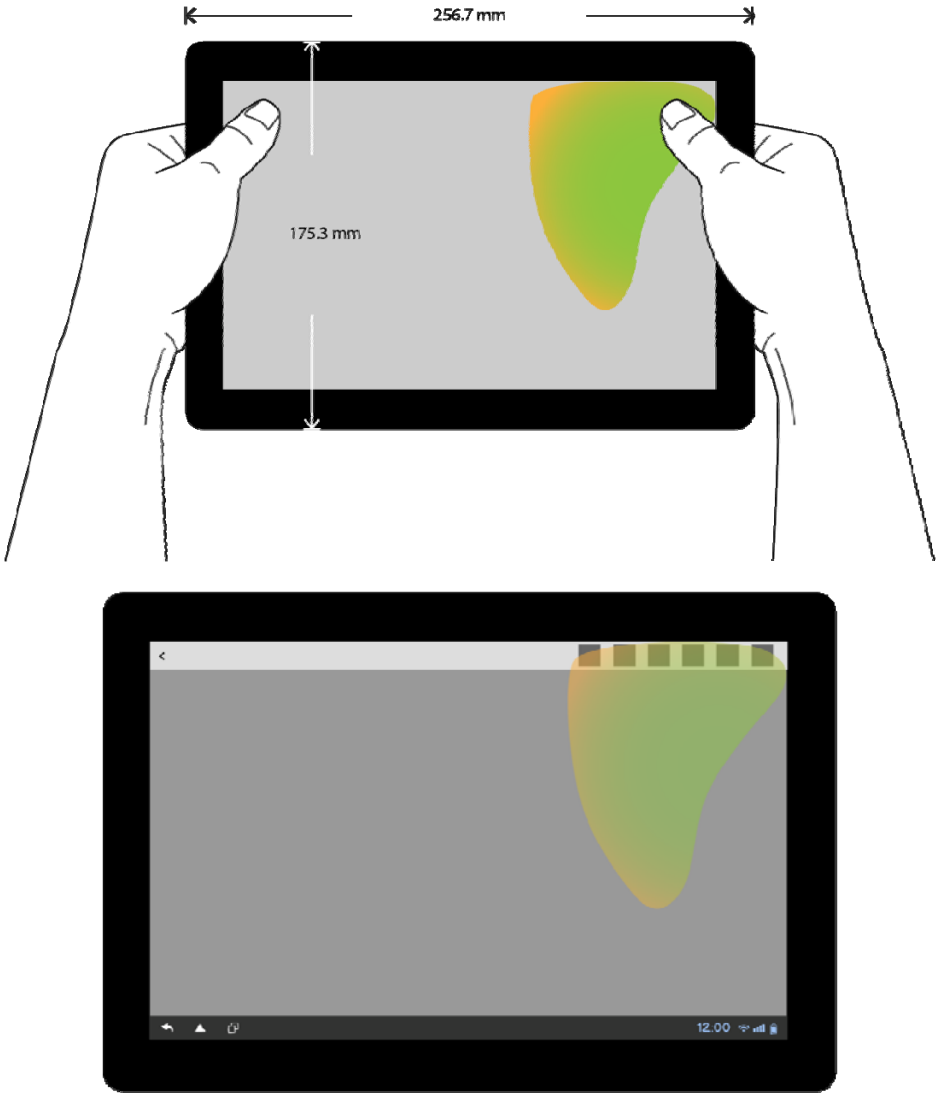


Figure 21 Touch comfort zone for an user holding a 10'' tablet PC with both hands.

However, the guideline suggested action buttons placed on the action bar shall not be more than five, and extra shall be placed in action overflow. In our holding tests, testers felt no difficulties in clicking the 6th action button counted from right to left, with their right thumb while holding the 10.1'' tablet computer (See Figure 21).

7.5.2.2. ActionButtons

For creating action buttons, first of all we needed to make clear the user flow to prioritize actions. Android's guidelines also incepted the FIT scheme, namely Frequent, Important and Typical.

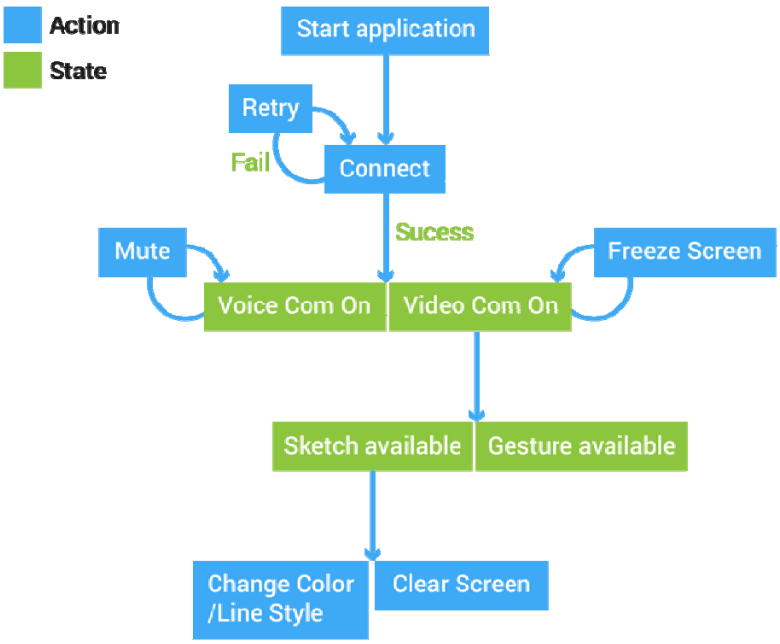


Figure 22 User flow

As the user flowchart shows (See Figure 22), after starting the application, the user need to build a connection to the server (helper) side. This action may repeat times if the server side does not answer the request or due to network failures. If the connection is built successfully, the user can receive audio signals and video images from the server side. Two actions then become available here, Mute and Freeze Screen. The Mute action is applied on voice communication and the Freeze screen action will pause the video streaming and keep the screen showing the last available image. While the video communication is running, whenever in play mode or the frozen mode, the states of providing sketch and gesture become available. The user can make actions at the time to change the color and line style of the sketch, or clear all the existing sketches on the screen.

In the first functional prototype, with the concern on the following user-test, we have to set video on/off as an action in the application. Therefore, we got a list of actions based on the user-flow, such as Connect, Mute, Turn on/off video chat, Freeze screen, Turn on/off gesture instruction feed, Change color and line style and Clear screen.

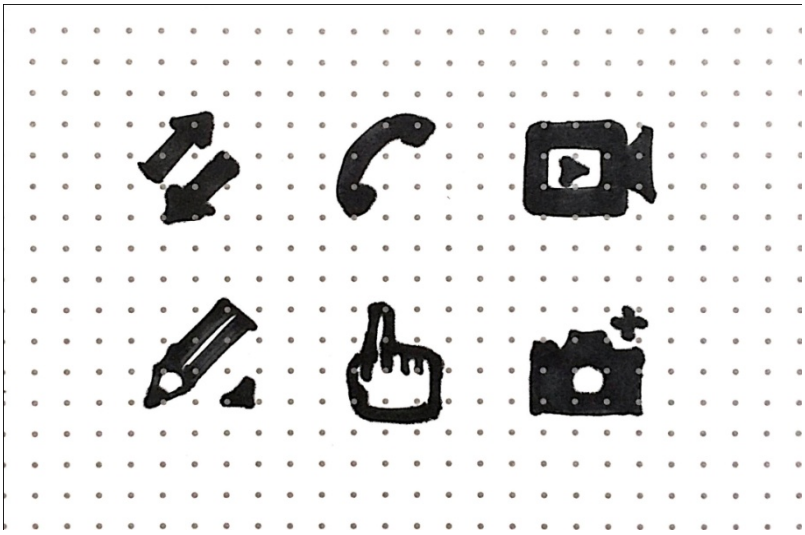


Figure 23 Design of button-icons.

When prioritized the actions, we took Change color and line style and Clear screen as minor actions, which are not expected to be used by users, and put these two action buttons under a ‘sketch/pencil’ spinner. However, in the following user-test, because helpers felt so intuitive to provide sketch instructions, drivers in our tests always needed to clear their screen and were not able to find the Clear screen action button efficiently. This fact shown in tests indicated problems in the GUI design might lead to unexpected user-experience and we must fix those in the further research.

7.5.2.3. Dialogs

As the Android guideline suggested, dialogs only happen when the user need to confirm a choice, or to make complex input. For this reason, dialogs in our driver-side application only show in the configuration of connection setup and color settings.

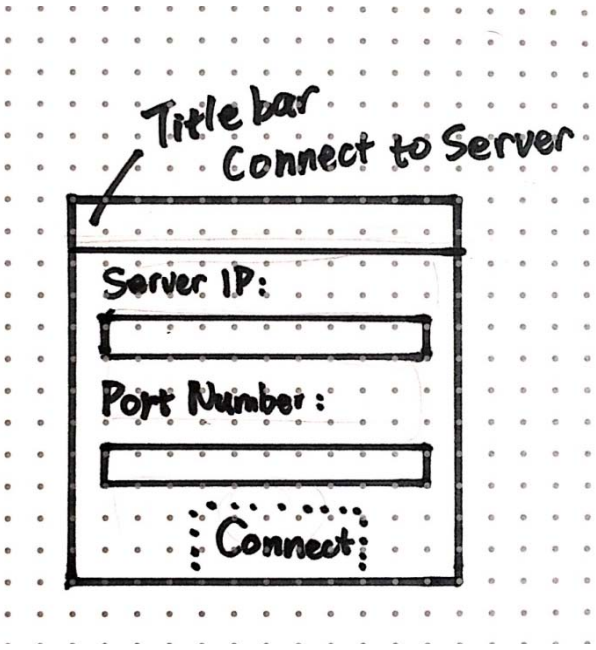


Figure 24 Wireframe of network connection dialog.

The dialog window of connection configuration contains three main sections as well, namely the title region, the content area and the action button. Our functional prototype only works in a local area network, thus users have to configure the IP address and port number of the server (helper side). There are two action buttons showing to the bottom. If the user has addressed both the server IP and the port number, the Connect button becomes available. Otherwise, the user can click Cancel button to abort the dialog window (See Figure 24).

The color setting contains two parts. The user can change the color and the transparency of his/her next sketch. With making sketches in different color, the user can denote different objects on the screen and differentiate them at a glance. (See Figure 25)

Change the transparency of the sketch tool is also quite helpful for drivers to have more than one way to make notes. The user can set the sketch to non-transparent to make notes or draw lines. In contrary, he/she can also set the transparency between 1%-99% to highlight objects on the screen. (See Figure 26)

Figure 25 Changing color of sketch to denote objects.

Figure 26 Changing opacity of sketch to highlight objects.

In order to achieve said settings, a color picker is designed to be placed in the upper part of the color setting dialog, and a scroll bar for changing transparency stays in the lower part of the window. When the user finishes all the changes on color, he/she can click the action button below to save the setting. (See Figure 27)

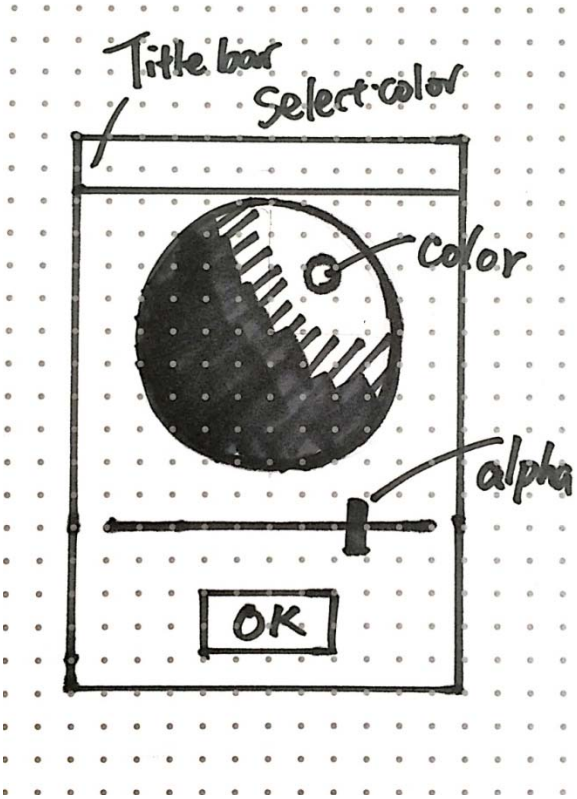


Figure 27 Wireframe of color setting dialog

8.System Evaluation

We carried out a user-study in order to assess the usability, functionality, and performance of the system. The tasks and test environment were designed to be similar to car repair work and its associated environment. The goal of the system evaluation was to find out whether the new functionalities, such as sketch overlay and gesture overlay, could improve the quality of support provided by the helper. The user-study of our system not only assesses the system's usability and functionality, but also tries to make a comparison between the new solution and the voice-only tele-assistance solution, which is already used by many companies. We had two different scenarios in the user study. One condition is the SEMarbeta system with video streaming, sketch overlay, and gesture overlay. The control condition is voice-only, which works on the same hardware SEMarbeta setup, but without sketching and gesture overlays.

8.1.SEMarbeta vs. voice-only condition

The SEMarbeta condition was carried out in a wireless local area network. For the study, the driver subject worked on a Volvo V70 car while holding a Samsung Galaxy Tab 10.1 tablet PC. The SEMarbeta application for Android ran on the tablet and allowed the driver to start a video-call, transmit duplex painting information, and receive gesture information. On the other side, a helper subject sat in front of the stationary computer with the SEMarbeta application running. The hardware consisted of a PC, an Acer 23" touchscreen monitor, and a fixed camera sitting on top to capture the helper's deictic gestures and operations (See Figure 6).

In the voice-only condition, we use the same hardware setup as for the SEMarbeta condition, so that the voice-only condition had the same quality of speech transmission and portability of devices. Since the VoIP functionality and video streaming functionality were separated in our design, the voice-only condition could be achieved by turning off the video streaming functionality on the tablet the driver side. Thus, the helper side cannot get any video images even if the driver side can still see the workspace through the camera.

8.2.Subjects

A total of sixteen subjects (9 males and 7 females) took part in the evaluation. All subjects hold a valid driving license. Thus, we assumed that all subjects have a basic knowledge of automotive issues. For each round of the user-study, one subject acted as a driver who has to fix some problem at the car, while the other subject acted as a helper who provided support. In total, 8 groups took part in the evaluation. Subjects on the helper side were instructed to act as experts who can provide support at a professional level. In the subject-recruiting process, we found it difficult to find actual experts in automotive repair. Thus, all the helper subjects got to use an operation manual that holds enough information for troubleshooting and fixing the relevant problem, thus enabling them to act like real experts.

8.3.Experimental Design

The tasks in the user-study are designed to be similar to a real scenario in daily life. Since all subjects in the evaluation are not professional in repairing cars, we had to

provide basic tasks, which can be finished by normal drivers with instructions or manuals. In the official manual of the test car, the manufacturer provided some instructions on basic works, such as changing tires or examining fuses, changing the battery, checking the engine oil, and installing a child seat. However, we eliminated the task of changing wheels because it is dirty and may take more than half an hour to complete the work, as well as changing the battery because of safety issues.
For counter-balancing the learning effect, the kind of task was changed in each round. This required that the two tasks examined in the user-study had to be approximately of the same difficulty level. Hence, the two task chosen were i) checking oil engine compartment and ii) localizing the fuse for the rear audio system (See Figure 28). The two tasks are comparable in terms of complexity and difficulty, and also have three steps in common:
Step 1: Opening a closed container.
Step 2: Locating the right component.
Step 3: Reading and confirming a value (or: state).

Figure 28 Systems evaluated: SEMarbeta used for checking oil in the engine compartment (top) and voice-only used for examining rear fuses (bottom).

8.4.Procedure

Before starting a task of the user-study, a short task description was given to both subjects, and the ‘driver’ got an additional list with task requirements. Both subjects were told that they have to accomplish one task through voice-only communication, and the other task with our SEMarbeta system. Then, the two subjects received short instructions on how to use the SEMarbeta system. The helper side was introduced to the features of the sketch overlay and the gesture overlay, while the driver side learned the process of starting a video support call. In addition, the user manual of the

test car was handed out to the helper in order to help him to solve the driver's requests easily.

The driver went to the car in the garage where the helper cannot see or hear him directly without communication devices. At the beginning of each test round, the supervisors started the voice-only call or the SEMarbeta video streaming, depending on the schedule, and then gave the devices to the subjects. In addition, the supervisors started a video camera to record the processes at each side, and so the video recording could provide the completion time of the test as well. After finishing the first task, the supervisors switched the system to the other mode. After the subjects took a short break, they started the next task.

Once both tasks were completed, the supervisors interviewed the subjects and asked questions related to the comparison between the voice-only call and the SEMarbeta system, the user experience of SEMarbeta, and their suggestions on the future demands as well.
The analysis of the data produced by the user study indicated issues related to performance and cognition. We note that all subjects appreciated the concept of SEMarbeta. However, as said above, only 16 subjects, making 8 groups, took part in our user study, which is too few to yield significant differences.

9.Results

We expected the completion time of automotive repair tasks with the SEMarbeta system to be shorter than the completion time when using voice-call only. Moreover, we would have liked to see that the sketch overlay function and gesture overlay function could support the users' cognition to some degree, so that helpers and drivers could utilize those functions to make the communication process more convenient.

9.1.Objective Measures

Since only a small number of subjects attended our user study, we could not get enough valid data to prove a significant difference on the time performance between the traditional voice-only assistance and SEMarbeta. According to the average completion time, SEMarbeta performed better on both tasks. However, as mentioned before, the variance between the data is quite high. It is partly caused by the difference in level of repairing skill among the subjects and the unsteady performance of our system affected by the test environment as well.

Although no significant difference could be found to imply that SEMarbeta can shorten the completion time compared to the voice-only solution, we could still find some positive outcomes from the average value. Except for the number of subjects, another important factor that influenced our user study was the environment. The user study took place in a basement garage, where the wireless network signal could easily be affected or blocked by other facilities. Moreover, the garage's lighting was not good enough for the camera of the tablet PC to capture clear images. In one round of the user study, a subject could not see the switch of the engine hood clearly, which prolonged the completion time of that round severely and pulled up the average completion time of SEMarbeta system to some degree. However, this also showed limitations of our system, which cannot be used at night or under poor lighting conditions.

9.2.Subjective Measures

After each round of the user study, we interviewed the two subjects of that round. The interview focused on examining the usability of the system. The questions were tailored to driver subjects and helper subjects. Table 1 presents all the questions and some representative given answers. Besides the positive feedback, other interviews raised some problems related to the system performance, the GUI of the SEMarbeta application, and the interaction using gesture input. As expected, subjects expressed their positive attitude towards having SEMarbeta as a remote support solution to take the place of the former telephone solution (100% positive). However, from the observation and the final interviews, it also turned out that subjects using the system the first time had some recognition problems on the additional overlays.

Table 1: Subjective results: All questions (left) and some representative answers (right) for driver subjects (top) and helper subjects (bottom).

Driver subject questions:	**Driver subject answers:**
1. Do you think SEMarbeta can provide better quality of help compared with voice call?	● Users may have different backgrounds, different languagesand different abilities to fix the car. The system can help people in such situations. ● It performed badly in the dark environment, but others are good, which can easily show something to the other side.
2. What do you suggest on the sketch function?	● It is easy to use, but it is difficult to indicate the position if the worker is moving. ● It is a little complex to select the eraser function, and only one color is available.
3. What do you suggest on the gesture function?	● I could see the helper's instruction with his finger pointing things out. ● I saw the gesture from other side, but it was too big. It could be more helpful if the image was smaller.
4. Would you repair your car by yourself if you have SEMarbeta system?	● I would like to use it if the problem is more complicated, such as electronic repairing. I can solve them according to instructions, rather than read manuals.
5. Which device would you prefer to run this system? Phone or Tablet PC?	● Technically, having a tablet is better for the bigger screen to see the instructions, but I will not buy a tablet just because of having this system in my car.
Helper subject questions:	**Helper subject answers:**
6. Have you helped your friend on fixing things via phone before?	● I used to help them on computer problems by phone call. It was hard to understand what the other side was doing.
7. Do you think SEMarbeta can provide better quality of help compared with voice call?	● Yes, it enables users to see more. Sometimes it is difficult to describe a thing with only oral explanations, since people always don't know how to call it. With this kind of system, I can point it out and verify the other's actions.
8. What do you suggest on the sketch function?	● I could paint something on the screen to show what I mean to the other side. ● It should have some transparent properties. Because sometimes the sketches from two sides may overlap.
9. What do you suggest on the gesture function?	● It is easier to point out than paint with fingers or the cursor. However, sometimes I forgot this function. It may take time to learn and accept it. The Interaction way is good and intuitive. ● Since it is the first time to use it, people may be confused by the hand showing on screen and consider it belongs to

	the other side.
10. Would you try to help your friends on fixing things if you have SEMarbeta system?	● If I know how to fix the things, I will use the system to fix the problem. For example, I can fix computer programs, point out the button that my father need to click and so on. Video streaming is a good feedback for instructors while helping others who do not have enough knowledge to fix the problem.

10.Discussion and Future Work

*10.1.*Result Discussion

In terms of objective measures among driver subjects (or: drivers), only 25% used the sketch overlay to outline objects on their tablet PC. Analyzing the answers given by drivers, we found three main reasons for not using this function. The most frequently reason given was that it is more comfortable to point to objects directly with a finger rather than outlining things on the screen. Another reason given was that the outlining would not work when moving the tablet PC. Finally, subjects reported that they would have wanted predefined drag-and-drop shapes to highlight things more easily.

In terms of objective measures among helper subjects (or: helpers), we observed a major difference in usage between sketch overlay and gesture overlay. That is, 88% of helpers used the sketch overlay to outline things for the driver. In contrast, only 25% of helpers provided gesture instructions to the drivers in a sufficient quality. Analyzing answers given by drivers, we found two main explanatory factors. Three helpers did not remember this function even though it was briefly introduced prior to task solving. Two helpers judged that video communication combined with sketch overlay would be sufficient to solve the problem.

Since only a few helpers used the gesture function, we asked them for potential reasons. Most subjects stated that sketching is a more common and natural way of interaction than gesturing. Some of them focused on the touch-screen while sketching, rather than moving their right hand to the black pad to provide gestures.

Gesture capturing techniques implemented directly in front of the screen, which can be seen in CollaBoard, might have scored better than the solution we chose. Such gesturing is fully synchronized with sketching, while SEMarbeta requires indirect pointing, as input and output spaces are separate and only coupled through the captured hand image shown onscreen. For example, indirect pointing could be avoided if a camera would capture the helper's interaction directly on the screen. Still, there are two drawbacks of direct gesturing. Firstly, the image segmentation required for such solutions has some limitations. Second, in-the-air gesturing may cause fatigue for professional helpers who have to work long hours.

10.2.Process Discussion

agile development and user test
the quality of prototype
Regarding the process of design and implementation in this master thesis project, we had learned a lot from what the result of user test has been reflected. There are three main issues considered to be the most remarkable points that we wanted to raised for the interaction design study.

The first noticeable issue in the process of the thesis project is the absence of agile development and several rounds of iterations. Since the concept of this project may require the implementation of not only an independent application but a collaborative system functioning on both tablet and desktop platforms, the lifecycle of our implementation took about two months with no enough time left due to the duration requirement of master thesis study. We believed many of the flaws in the first

prototype can be improved if a second iteration can be practiced. For example, in the first round of user test, we have found most of the users may prefer a one-hand handheld device in the real use scenario rather than a bigger screen, even the image of instruction can be clearer on the bigger one. However, due to the duration constraint, there was no time for us to fix those non-conceptual defects.

Besides the lack of iteration in the process, we also found the quality of software prototype may also effect the result of user test. In the last chapter, we have described the things happened in the user test such as the network stability has strong impact on the completion time of test-tasks. The fact was that the first version of hardware implementation and software prototype were only tested under the lab environment that provides stable network connection and wifi signals without interference. However, while in our user tests, our testers experienced some crash down or stuck of video images in the communication due to the loss of network signals, especially if the user is using the video chat instead of pure-audio communication, and these 'accidents' actually also led to a non-statical significant result of our test.

Finally, we would like to discuss an interesting dilemma of the design of user test in the interaction design study. From many pre-study literature on group collaboration and remote collaborative study, we found most of the proving session adopted simple and basic tasks as the assignments, such as assembling LEGO bricks or connecting the dots. Researchers believed the simplification of test tasks can reduce the interference factors and be easily controlled in some extend, and may get a 'good' test result. But in our practice of user study, we chose two real tasks that are close to the daily use cases, which are not that easy to be controlled due to multiple factors. For example, in one round of the user test, the tester who performed as a driver had great difficulty in finding the gripe for opening the hood, because he used to drive a SAAB instead of Volvo cars adopted in our test. This unexpected situation brought extra five minutes for him to complete the task and also effected our final statical result in a big degree since we have limited testers in the user test session. However, practicing real and virtual tasks in the user tests may also bring many good hints and experience to the concept, prototype and product. By using the same example in the last paragraph, without a real test environment or a similar setup which is close to the real use scenario, we could not realize the importance of network stability and anti-jamming in the remote collaboration products. For another instance, when the user performed the diagnosis tasks, they had to use both hands or one hand to hold the communication device with another doing the task. In such situation, the setup of our concept of using tablet for presenting instruction was not that pragmatic. But those facts we would never learn if we adopted some idealistic, simple and well-controlled tasks.

10.3.Generalizability and Future work

In future work we plan to improve information presentation, thereby aiming to reduce cognition load. Current gesture presentation is still quite limited by the segmentation algorithm and image processing ability of mobile devices. We believe the interaction concept of Kinect can be a way to realize gesture recognition in front of the screen. The concept of capturing body movement to control the movement of a 3D skeleton and then rendering the skeleton to a 3D avatar could bring two benefits to our system. From the user study, we learned that helpers always make gestures during their communication, even though they knew their gestures could not be captured by our system. It is a natural response when helpers meet some difficulties in speaking out

the exact word they want to describe the operation. If a device similar to Kinect were used in our system, the helpers could provide their gestures more naturally during the support process, with no limitations from the system on providing gestures in some fixed area. Moreover, when using a Kinect instead of the normal image capturing process, hand gesture recognition is no longer limited by ambient light conditions or background images. (See Figure 29)

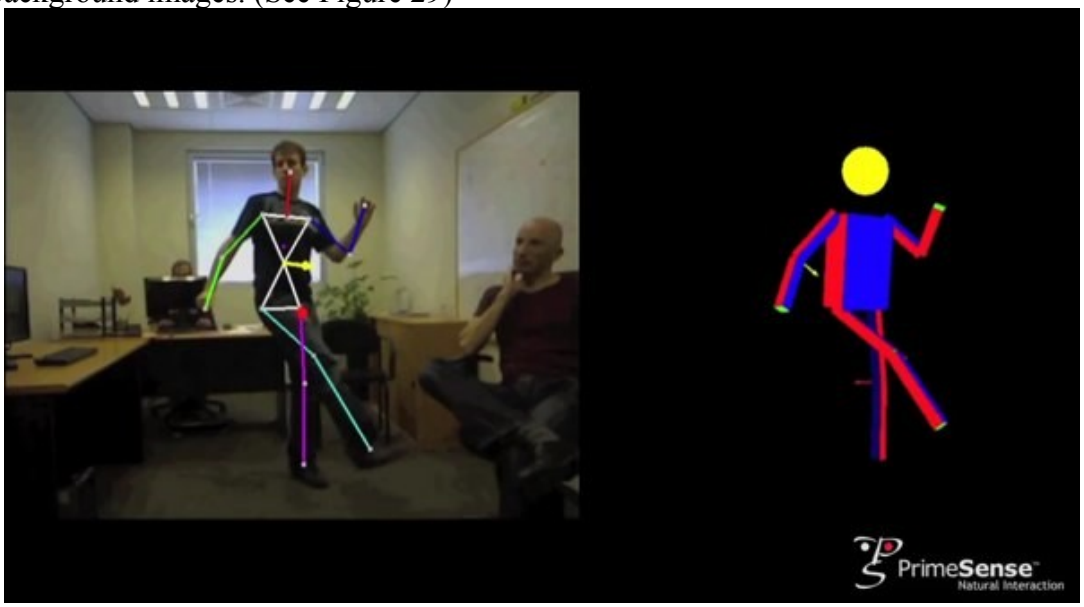


Figure 29 Variation of Kinect

However, even the Kinect has a high-speed camera with inferred sensors as well, which changes the interaction from traditional photography to so called In-air control. The standard Kinect technology can only recognize the motion of body parts. Some hacks changed the algorithm of Kinect device and made it recognize finger movements already, but the accuracy and depth recognition of Kinect still cannot fulfill the requirement of providing operational gestures that have more details on finger movements.

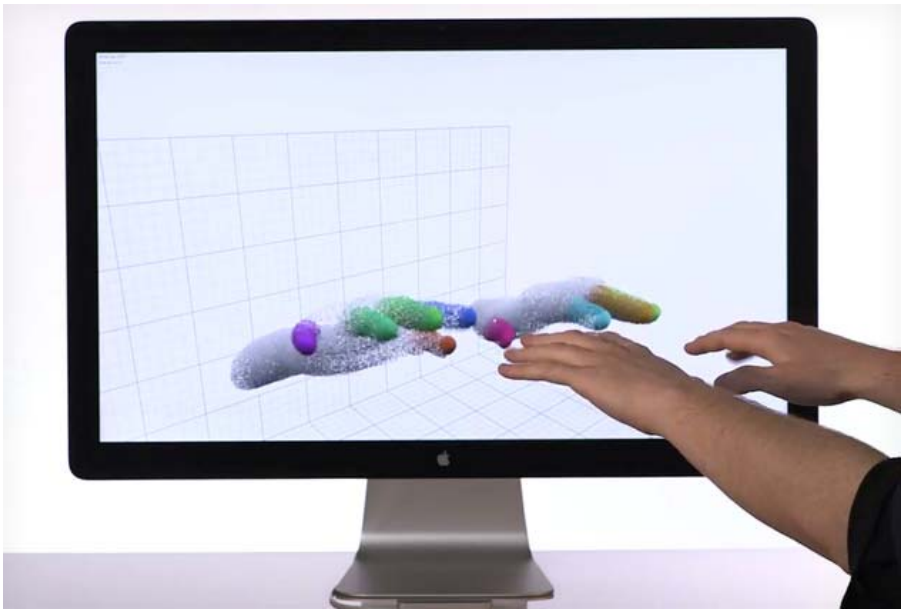


Figure 30 Leap motion 3D demo

In the near future, Leap 3D can be an idealistic device for our concept implementation. Leap Motion can track multiple objects and gestures in detail, with a really high accuracy to 0.01 mm, which is about hundreds times more accurate than the Kinect. When it starts, it can generate a 3D recognition space of about 4 cubic feet. From the image, it seems adapting some kind of scanning technology to build a global view of the objects that in the space. The algorithm rebuild the object from the enormous dots sensed, rather than using point-tracking. However, the high accuracy and really cheap price also bring big controversy that some experts think it is too good to be true as the Leap's marketing. (See Figure 30)

References

[1]. Alem, L., Tecchia, F. Huang, W.: Remote Tele-assistance System for Maintenance Operators in Mines, In: 11th Underground Coal Operators' Conference, 2011, 171-177.

[2]. Tang, A., Neustaedter, C., Greenberg, S.: Videoarms: embodiments for mixed presence groupware. In: People and Computers XX—Engage, pp. 85–102 (2007)

[3]. Kunz, A.; Nescher, T.; Küchler, M.: CollaBoard: A Novel Interactive Electronic Whiteboard for Remote Collaboration with People on Content; in: Proceedings of the 2010 International Conference on Cyberworlds – CW 2010; pp. 430 – 437; 20. – 23. October 2010; Singapore

[4]. http://www.arvika.de/www/index.htm (accessed 1.5.2012)

[5]. Papadopoulos, C.: Improving Awareness in Mobile CSCW. In: IEEE Transactions on Mobile Computing, Volume 5 Issue 10, October 2006, 1331-1346

[6]. O'Hara, K., Black, A. Lipson, M.: Everyday Practices with Mobile Video Telephony. In: Proceedings of the SIGCHI 2006 Conference on Human Factors in computing systems, 871-880

[7]. Herskovic, V., Ochoa, S. F., Pino, J. A.: Modeling Groupware for Mobile Collaborative Work. In: Proceedings of the 2009 13th International Conference on Computer Supported Cooperative Work in Design, 384-389

[8]. Spikol, D., Milrad, M., Maldonado, H., Pea, R.: Integrating Co-Design Practices into the Development of Mobile Science Collaboratories. In: Proceedings of the 2009 Ninth IEEE International Conference on Advanced Learning Technologies, 393-397

[9]. Anastassova, M. Burkhardt, J.M.: Automotive technicians' training as a community-of-practice: Implications for the design of an augmented reality teaching aid. In: Journal of Applied Ergonomics. 2009 Jul; 40(4): 713-21.

[10]. Ran, Y., Wang, Z., Zhu, F.: Trends of mixed reality aided industrial design applications. In: Proceedings of 2011 International Conference on Energy Systems and Electrical Power (ESEP 2011), 3144–3151

[11]. https://plus.google.com/111626127367496192147#111626127367496192147/posts (accessed 18.5.2012)

[12]. http://www.samsung.com/ch/consumer/mobile-phone/tablets/tablets/GT-P7500UWDITV; accessed 18.5.2012

[13]. http://research.microsoft.com/en-us/projects/conferencexp/; accessed 18.5.2012

[14]. http://en.wikipedia.org/wiki/User_Datagram_Protocol; accessed 18.5.2012

[15]. http://en.wikipedia.org/wiki/Voice_over_IP; accessed 18.5.2012

[16]. McKay H, Dudley B. About storytelling: A practical yguide [M]. Hale & Iremonger, 1996.

Appendix A

"SEMarbeta: Mobile Sketch-Gesture-Video Remote Support for Car Drivers"

A publish paper in 4th Augmented Human International Conference (AH'13)

SEMarbeta: Mobile Sketch-Gesture-Video Remote Support for Car Drivers

Sicheng Chen [1, 3], Miao Chen [1, 3], Andreas Kunz [2], Asim Evren Yantaç [3], Mathias Bergmark [1], Anders Sundin [1], Morten Fjeld [3]

[1] Semcon Human Factors
Theres Svenssons gata 15
SE-417 80 Gothenburg
sicheng.chen@semcon.com

[2] ICVR
ETH Zurich
CH-8092 Zurich
kunz@iwf.mavt.ethz.ch

[3] t2i Interaction Lab
Chalmers Univ. of Technology
SE-412 96 Gothenburg
morten@fjeld.ch

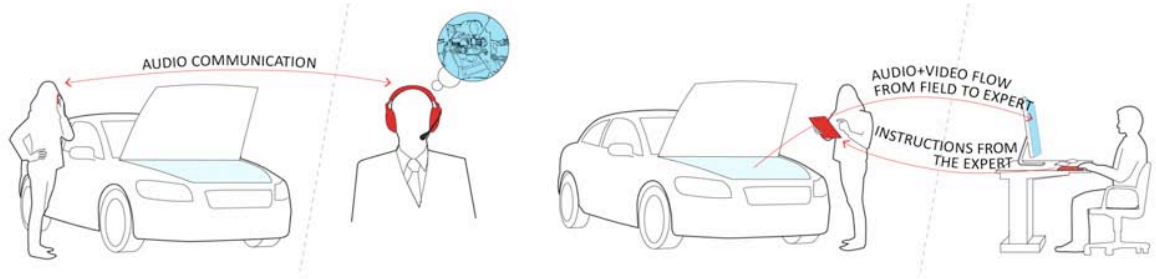


Figure 1: Remote support for car drivers is typically offered as audio instructions only (left). This paper presents a mobile solution including a sketch- and gesture-video-overlay (right).

ABSTRACT

Uneven knowledge distribution is often an issue in remote support systems, creating the occasional need for additional information layers that extend beyond plain videoconference and shared workspaces. This paper introduces SEMarbeta, a remote support system designed for car drivers in need of help from an office-bound professional expert. We introduce a design concept and its technical implementation using low-cost hardware and techniques inspired by augmented reality research. In this setup, the driver uses a portable Android tablet PC while the expert mechanic uses a stationary computer equipped with a video camera capturing his gestures and sketches. Hence, verbal instructions can be combined with supportive gestures and sketches added by the expert mechanic to the car's video display. To validate this concept, we carried out a user study involving two typical automotive repair tasks: checking engine oil and examining fuses. Based on these tasks and following a between-group (drivers and expert mechanics) design, we compared voice-only with additional sketch- and gesture-overlay on video screenshots measuring objective and perceived quality of help. Results indicate that sketch- and gesture-overlay can benefit remote car support in typical breakdown situations.

AH'13, March 07 – 08 2013, Stuttgart, Germany.

Categories and Subject Descriptors

H.5.3 [**Group and Organization Interfaces**]: Computer-supported cooperative work, Evaluation/methodology, Synchronous interaction

General Terms

Performance, Design, User study, Human Factors, AR

Keywords

Remote support, automotive, mobile, handheld computer

1. INTRODUCTION

While troubleshooting in modern vehicles can be a challenging task for professionals, the average driver has even more problems working with automotive technologies. Even though most technical details are described in the car's handbook, many drivers are overwhelmed when having to diagnose and fix even simple problems in their car. For instance, even minor issues, such as a blown fuse, may turn out to be expensive if the driver has to call for service personnel to come and fix the problem. Physical presence of service personnel is required whenever audio communication channels such as mobile phones are insufficient. Even when audio instructions prove to be sufficient, drivers might not understand the instructions and therefore be unable to solve the problem themselves. Car drivers increasingly carry smartphones, tablets, or other devices made for sketching and video streaming. These communication channels could enable service personnel to offer help in troubleshooting without their physical presence (Fig. 1).

This paper presents remote video support technology that allows a) transferring sketch-overlaid video screenshots from the driver

to the expert mechanic (the helper), and b) sending back sketch- and gesture-overlays on still images from the remote helper to the driver. While live video is streamed from the driver- to the helper-side, the driver will typically freeze the video to perform sketching. While our work is inspired by insights from Augmented Reality (AR), here we explore the use of widely used standard mobile devices. With such set-up, we design a remote instructor-operator collaborative system to support minor automotive breakdown cases.

The next section offers related work, followed by a section on system concept and design. The fourth section describes hardware implementation and the fifth software implementation. The sixth section offers system evaluation, and is followed by a section presenting a discussion and an outlook on future work.

2. RELATED WORK

With the "spread of wireless communication and the desire to travel 'light,' collaboration across mobile devices", such as phones, tables, and notebooks, is a likely trend for future groupware applications [3]. With the diffusion of mobile devices in the working landscape, there is an economic interest in using standard mobile devices to develop remote expert support [12]. A video image may even reinforce the affective experience in communication between geographically separated instructors and operators [11]. To model the complexity in mobile groupware systems, a graphical language describing loosely coupled work patterns has been suggested [5]. In the educational field, researchers have proposed solutions for collaborative learning outside the classroom [16]. While our work will target the use of standard devices in a mobile remote expert setting, it will not yet involve the use of AR techniques or the use of a Head-Mounted Display (HMD). Nonetheless, we find it instructive to start with a review of a few AR-related projects that laid foundation for understanding remote computer-mediated instructor-operator collaboration.

Early AR-related works on remote instructor-operator collaboration were presented in SharedView [8]. The operator wears a so-called shared camera and a HMD. The shared camera follows the operator's view and thereby jointly shows the task and the operator's field of view to the instructor. The system also transmits overlaid gestures in both directions. Extensive experiments with SharedView showed that to assure high user acceptance and effective use, it is important that the "system is an extension of an instructor's body" and that users' acceptance is high [8]. Such insights paired with observations from face-to-face instructor-operator collaboration were later used to refine design requirements. Hence, in the GestureCam project, the authors suggested either using a second camera that the instructor can control remotely, or widening the camera's field view so that many objects can be seen in the display at the same time [9]. In a more recent remote instructor-operator collaboration study, shoulder-worn active camera and laser (WACL) was compared with traditional HMD [7]. The authors showed that WACL was superior to HMD in several ergonomic aspects such as comfort and fatigue. In a follow-up study, the same team compared a WACL-HMD combination with a WACL Chest-Worn Display (CWD) combination to examine which form of visual assist is most suited for the WACL [14]. The authors found that the CWD is superior to HMD and showed that the WACL can give improved task performance when paired with a worn display. Recently, there has been an interest in identifying and tracking unknown features in an unprepared environment. Such model-free markerless tracking was examined for remote support and it was shown that the sole requirement to an unknown scene is the presence of locally planar objects [10]. Finally, Fussel et al [4] examined tools supporting "remote gesture in video systems being used to complete collaborative physical tasks-tasks" in which two or more subjects work together manipulating three-dimensional objects in the real world. They studied pointing gestures, representational gestures, and how to support erasing gestures.

Besides the AR-related issues of display, tracking, and pointing techniques, recent research into remote instructor-operator collaboration has been geared towards particular domains of application. In ReMoTe [1], for example, remote instructing for miners working underground is targeted. The helper side of ReMoTe can get a visual understanding of the working situation by viewing live-video captured by the worker's head-worn camera. Meanwhile the system also captures helpers' instructions from their display and sends them back to the workers' side. HMD- and marker-based AR has been examined for automotive assembly, inspection, and repair [19]. Also within automotive assembly, guidelines for AR-based training were suggested [2]. Widening the focus to design at large, AR can also be a meaningful technique, as it may help designers to express innovative ideas and overcome technical difficulties [13].

Group awareness is another aspect of remote collaboration. Ways to enhance awareness between remote workers has been presented in VideoArms [17] and in CollaBoard [6]. These systems employ live-video gestural overlay on shared workspaces, thereby giving remote workers a side-by-side impression. While VideoArms shows collaborators hands and arms, CollaBoard transmits the image of the collaborator's upper body.

3. SYSTEM CONCEPT AND DESIGN

The process of reaching a system concept and design consisted of four subsequent steps. First, we conducted a participatory design process. Second, the insights first gained from participatory design were intersected with insights drawn from related work. Third, hardware architecture was laid out and defined. Fourth, we investigated vision-based gesture capturing strategies.

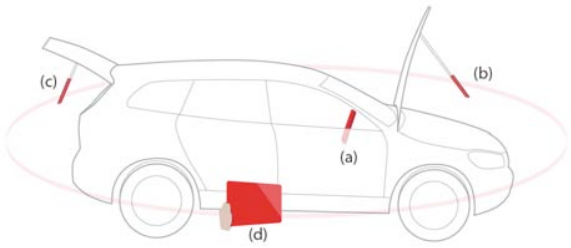


Figure 2: Our participatory design workshop indicated that mobile devices would have to be multipurpose. For example, the same device should offer a) infotainment functions, support b) engine, or c) luggage compartment instructions, and d) potentially instruct drivers how to change tires.

3.1 Participatory Design

Since the research was done in collaboration with an automotive design company, we employed a participatory design approach. Senior engineers from the company presented requirements and provided professional suggestions based on their long experience on automotive design and information presentation. Hence, we developed the idea of a system that leverages mobile devices to

offer remote technical support for car drivers. We researched existing remote support services in the automotive field, as well as automotive support applications provided in iOS App stores and in the Android market. We found that the envisioned remote support system for drivers could have a major potential since the current solutions all depend on voice-only support. While video is offered as part of pre-recorded repair instructions, such products are neither interactive nor adapted to current breakdown situations. We also found that customers in the automotive industry tend to prefer multipurpose solutions where mobile devices not only offer remote support but also provide remote control for other car functionality such as an infotainment system.

In the process of gathering requirements, we used a so-called scenario method to retrieve customers' needs [19]. We produced a short video showcasing a typical use of our envisioned system; it shows a driver facing an engine breakdown. She picks up her mobile device and starts an app provided by the car manufacturer. After having connected with the helper side, the expert mechanic helper provides instructions using sketching and gesturing. Based on this demonstration it became easier for the senior engineers to understand the overall concept of our system, and thus be able to suggest relevant design goals, such as reducing the cost of use of the helper side, and defining the handheld device as a future standard automobile feature (Fig. 2).

3.2 Related Work—Design Specs Intersect

The concept of the ReMoTe system is similar to our research objective, in which the helper side can provide additional information to the worker side that is not limited to simple linguistic instruction. However, the work of conceptualizing, designing, and implementing equipment for professional users is not the subject here. That is, while HMD, head-worn camera, and laser pointer may work for professionals in ReMoTe, SharedView, or GestureCam [1, 8, 9], such devices are most likely not apt for an average end-user of mobile devices. Moreover, for cases where the HMD is not a see-through device, combining hand images with live video becomes even more intricate. Google's glass project [22] might be a future device from which our envisioned system could conditionally benefit [18].

Other AR solutions in the automotive field, like ARVIKA [21], depend on pre-defined fiducial markers for position track and 3D animation. This kind of solution is not suitable for remote support, since everyday real-time diagnosis and instruction takes place in an environment without visual markers [13].

Furthermore, VideoArms [17] and CollaBoard [6] inspired our system concept. In CollaBoard, the full upper body of a worker is displayed on the remote side. Thus, all information like postures or deictic gestures is in context with the underlying content of the whiteboard. Although we do not need postures, transferring of deictic gestures is crucial for our system, since it is the most natural explanatory gesture.

Finally, since we want to present a mobile solution for the driver side, an important design aspect is device size. While the device should have a screen large enough for the driver to unequivocally recognize the helper's gestures in relation to the underlying image, it should still be a handheld portable device.

3.3 Hardware Architecture

Our system consists of a handheld device (tablet) and a stationary computer (PC) (Fig. 3). Both the tablet and the stationary computer offer sketching and audio communication. The driver side can transmit live video streams to the helper side in order to describe the problem (e.g. check oil, locate fuse). The helper receives live video streams from the remote situation (e.g. the car engine or fuse board) and can give back instructions on how to check oil or fuses, either by outlining directly on his screen or by using gestures that are captured by a camera. His gestured instructions and sketched outlines are offered as separate layers directly overlaid on the still image. Using another color for sketching, the driver can also outline to clarify problems.

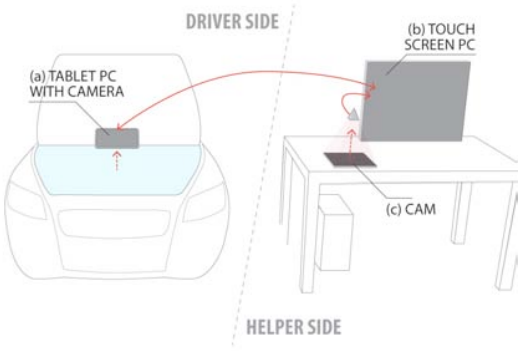


Figure 3: SEMarbeta hardware architecture

3.4 Gesture Capturing Strategy

Gesture capturing in front of a highly dynamic background such as a live video is a delicate task. While VideoArms used color segmentation algorithms to capture the deictic gestures in front of a screen, CollaBoard used a linearly polarizing filter in front of the camera and thus benefitted from the fact that an LC-screen already emits linearly polarized light. However, each method has specific shortcomings. While color segmentation only works well if no skin-like colors exist on the screen, the solution with polarized light cannot detect dark objects unless they differ significantly from the dark-gray of the captured image of the screen.

Although our system could be adapted to both of these segmentation strategies, we adopted another solution to capture the helper's gestures. We did not use any polarizing filter nor color segmentations. Instead, we mounted the camera next to the stationary computer facing downwards towards a black mat (16"x12") on the table. Thus, the helper can put his hand between the camera and the mat for his gestures to be captured. Here, we make use of the fact that we are used to this spatially distributed interaction. With a mouse, we regularly control the cursor, click a button, or draw a line while keeping our eyes on the screen (Fig. 4, bottom). However, there are some limitations to such gesture capturing that we will discuss in later sections.

4. HARDWARE IMPLEMENTATION

We set out to employ inexpensive hardware only. We also took into account that a driver's mobile unit should be low-weight for ease of handling and should be usable for other tasks. Unlike VideoArms or CollaBoard, we propose an asymmetric setup while maintaining several CollaBoard features. Two sides are involved in our system, helper side and driver, but each side works in a different context (Fig. 3). This partly asymmetric system set-up is presented next.

4.1 Helper Side

On the helper side, streamed or captured images from the driver side are combined with detected sketching (Fig. 4). Thus, standard touch screens are sufficient. In the prototype, a 22" touch screen is used. Since we use a touch screen, no further input devices, such as mouse or keyboard are needed, and the helper can easily use a pen or finger to interact with the software. However, since gestures should also be captured and transferred, an additional input capability is required.

Like in CollaBoard and VideoArms, the SEMarbeta system captures the helper's deictic gestures using a camera. As for the cost and quality of such a camera, we selected a high-resolution webcam with an auto-focus function for our prototype (Logitech QuickCam Vision Pro 9000). For our prototype, there was no need to apply polarizing filters to eliminate the background image of an LC-screen, since our setup was different to CollaBoard and VideoArms. This is further discussed in the software design section below.

No specific setup for the audio channel was required. The default microphone in the camera is used for audio input. For clearer audio quality, headphones are connected to the audio output. The helper side application is running under Windows 7 OS. Since the application runs an image processing function as well as a video transmission, a powerful CPU (Intel i5 CPU) is required. The computer is connected to the LAN.

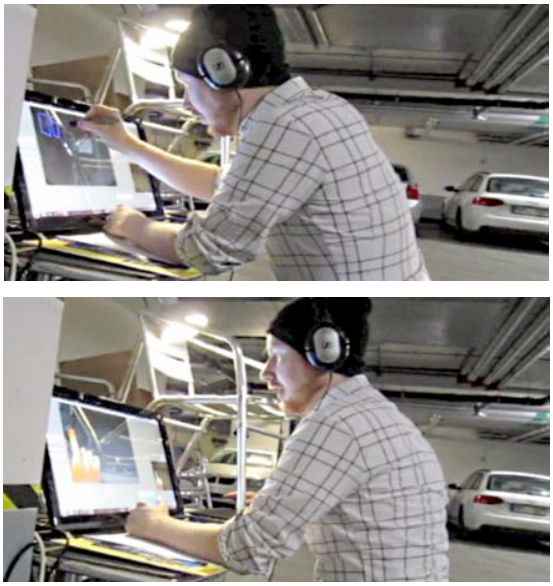


Figure 4: Implementation of the helper side: Sketching (top) and input of deictic gestures where the helper's right hand is captured and overlaid on screen (bottom).

4.2 Driver Side

In a mobile setting, a user must be able to pick up and hold a device easily and conveniently. For our application, the device has to run an image processing in order to guarantee a smooth video transfer. Here, we chose a Samsung Galaxy Tab 10.1 [23]. This product provides multiple network connections (Wi-Fi and 3G), so that the driver can connect at any time as long as a network is available. The device also has two cameras: one in front and on the back; the back camera is used to capture breakdown situations, while the front one is left unused.

5. SOFTWARE DESIGN AND IMPLEMENTATION

To our knowledge, Microsoft does not support the ConferenceXP [24] remote presentation software anymore (as it was used in the CollaBoard system). Therefore, new software was developed for our prototype, which can be used for the remote support system (Fig. 5). This software provides three different information layers at each side (audio, sketching, and image capturing). In our software architecture, the driver will use Layers 1-3, while the helper will use Layers 4-6. Next, we describe implementation and functionality of our software running on both helper and driver sides. We also present and analyze our implementation of user interaction.

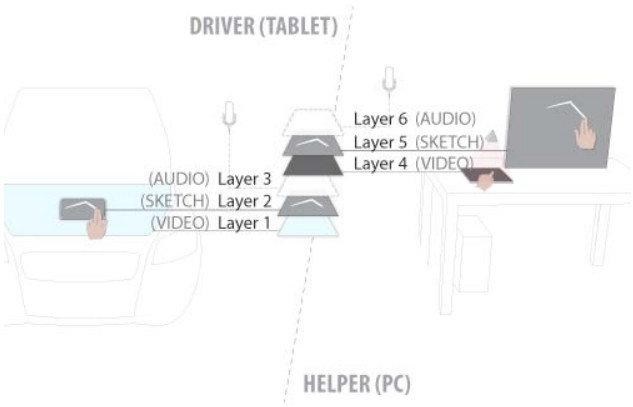


Figure 5: SEMarbeta software architecture

5.1 Audio and Video Connection

In order to realize a smooth video and audio transfer, the UDP [25] protocol is adopted for the transmission. A driver with a technical problem in the car can directly start a VoIP [24[] call to the helper side, while the helper can decide whether to accept the call or not. When accepted, audio is first transmitted to the other side. After the helper and the driver have established the initial communication, they can activate a live-video stream in case the helper thinks the problem is too difficult to be explained by voice-only, or if the driver considers the problem too difficult to describe. In this case, the Samsung Galaxy Tab transmits a video or a still image of the working scene to the helper side.

5.2 Screenshot Functionality

The reason for having the screenshot functionality is obvious. In our use case, the driver has to hold the device on one hand while performing the repairs with the other. Live video would result in a very unsteady image, which is not suitable for sketch or gesture overlay. In our design process, we observed that it is difficult for both the driver and the helper to point at a certain object or to outline an object on live video. Even the slightest movement of the device would disturb the analysis of the problem by the helper and thus hinder the discussion. Consequently, we designed and implemented screenshot functionality in the driver side application, where the Android tablet can temporally freeze the screen, so that the helper and driver can discuss the issues based on still image.

5.3 Sketch Overlay

The sketch overlay we implemented is one of the essential functions in our system. When the helper and the driver collaborate, it is difficult for the helper to explain technical issues by audio only, even when the helper is fully aware of the problem. This can mainly be ascribed to an uneven helper-driver knowledge distribution. Since sketches can help towards shared understanding, we developed the sketch overlay. It has basic painting tools for both the helper and the driver, and allows outlining the issues directly, which will be subsequently transmitted to the other side (Fig. 6).

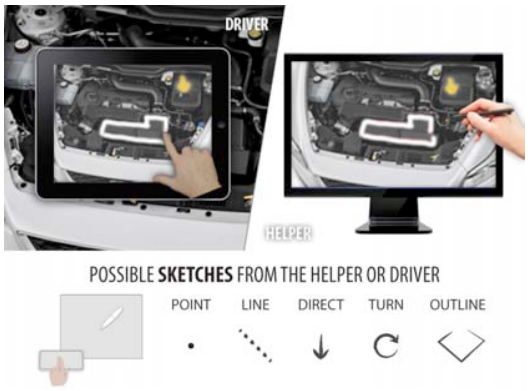


Figure 6: Two-way sketching function (top) and potential sketches for use by the driver or helper (bottom).

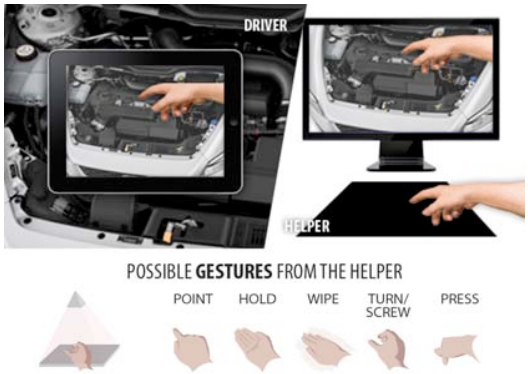


Figure 7: One-way gestures capture and overlay function (top) and potential gestures for use by the helper (bottom).

5.4 Gesture Overlay

While gesture capturing implemented has a distinct layer only on the helper side (Fig. 7), gestures are captured and shown on both sides. When a troubleshooting strategy is hard to explain, using hand gestures may help in clarifying the situation. However, deictic (e.g. "this handle" or "that fuse") gestures are only relevant when shown in relation to the problem, that is, the underlying image. To capture a hand gesture, but not the local background, it must be segmented from the background. We chose an image processing function where the software captures an image of the hand in front of a unique black background. A gray-scale function is then used to transform the whole image into gray-scale image. After this, a mask function converts the gray-scale image into a mask image, depending on a specific threshold value. Pixel gray-scale values below the threshold are set to 0 (black); values above are set to 255 (white). Since the background of the gesture is a black mat, the gesturing hand is white against a black background within the mask image. In a next step, another function named 'processing' compares the original image with the mask image. If the color of pixel is black in the mask image, then the color of that pixel in the original image will become transparent. The final result of this segmentation pipeline is an image of the hand in full color against a transparent background, which will then be overlaid atop the still image from the driver side.

6. SYSTEM EVALUATION

We carried out a user study in order to assess the usability, functionality, and performance of the system. Two tasks and a test environment were designed to come close to a minor car breakdown situation in an authentic environment. The goal of the system evaluation was to find out whether the new functionalities, such as sketch overlay and gesture overlay, could improve communication, collaboration, and thereby problem solving. To assess the SEMarbeta system's capacity, we compared the system with an industry-standard voice-only assistance implementation. Hence, the first condition was the SEMarbeta system with video streaming, sketch overlay, and gesture overlay. The second condition was voice-only working on the same hardware as the SEMarbeta setup, but without sketching and gesture overlays. We hypothesized that task completion time for the two repair tasks would be shorter with SEMarbeta than when using voice-call only. We also hypothesized that the sketch and gesture functions would support helper-driver communication and thereby collaboration.

6.1 SEMarbeta vs. voice-only condition

The SEMarbeta condition was carried out in a wireless local area network. For the study, the driver subject worked on a Volvo V70 car while holding a Samsung Galaxy Tab 10.1 tablet PC. The SEMarbeta application for Android ran on the tablet and allowed the driver to start a video-call, transmit duplex painting information, and receive gestural information. On the helper side, subjects were presented with the SEMarbeta application running on a stationary computer. The hardware consisted of a PC, an Acer 23" touchscreen monitor, and a fixed camera sitting on top to capture helper-side deictic gestures (Fig. 4). In the voice-only condition, we use the same experimental setup as for the SEMarbeta condition, so that this condition had the same quality of speech transmission and portability of devices. Since the VoIP functionality and video streaming functionality were separated in our design, the voice-only condition could be achieved by turning off the video streaming functionality on the tablet at the driver side. Thus, the helper side cannot get any video images.

6.2 Subjects

A total of sixteen subjects (9 males and 7 females) took part in the evaluation. All subjects hold a valid driving license. Thus, we assumed that all subjects have basic knowledge of automotive issues. For each round of the user study, one subject acted as a driver who has to fix some problem at the car, while the other subject acted as a helper who provided support. In total, 8 groups took part in the evaluation. Subjects on the helper side were instructed to act as experts who can provide support at a professional level. In the subject-recruiting process, we found it difficult to find actual experts in automotive repair. Thus, all the helper subjects got to use an operation manual that holds enough information for troubleshooting and fixing the relevant problem, thus enabling them to act like real experts.

6.3 Experimental Design

The tasks in the user study were designed to be similar to a real-life scenario. Since all subjects in the evaluation were not professional in repairing cars, we had to provide basic tasks that could be finished by normal drivers with instructions or manuals. In the official manual of the test car, the manufacturer provided some instructions on basic works, such as changing tires or examining fuses, changing the battery, checking the engine oil, and installing a child seat. However, we eliminated the task of changing wheels because it is dirty and may take more than half an hour to complete. We also did not include changing the battery because of safety issues.

Figure 8: Tasks evaluated: SEMarbeta used for checking oil in the engine compartment (top) and voice-only used for examining rear fuses (bottom).

To counter-balance possible learning effects, the kind of task was changed in each round. This required that the two tasks examined in the user study had to be approximately of the same difficulty level. Hence, the two task chosen were i) checking oil engine compartment and ii) localizing the fuse for the rear audio system (Fig. 8). The two tasks are comparable in terms of complexity and difficulty; they also have three steps in common:

- Step 1: Opening a closed container
- Step 2: Locating the right component
- Step 3: Reading and confirming a value (or: state)

6.4 Procedure

Before starting a task of the user study, a short task description was given to both subjects, and the driver got an additional list with task requirements. Both subjects were told that they have to accomplish one task through voice-only communication, and the other task with our SEMarbeta system. Then, the two subjects received short instructions on how to use the SEMarbeta system. The helper side was introduced to the features of the sketch overlay and the gesture overlay, while the driver side learned the process of starting a video support call. In addition, the user manual of the test car was handed out to the helper in order to help him solve the driver's requests easily.

The driver went to the car in the garage where the helper cannot see or hear him directly without communication devices. At the beginning of each test round, the experimental leader started the voice-only call or the SEMarbeta video streaming, depending on the schedule, and then gave the devices to the subjects. In addition, the supervisors started a video camera to record the processes at each side, and so the video recording could provide the completion time of the test as well. After finishing the first task, the supervisors switched the system to the other mode. After the subjects took a short break, they started the next task. Once both tasks were completed, the supervisors interviewed the subjects and asked questions related to the comparison between the voice-only call and the SEMarbeta system, the user experience of SEMarbeta, and suggestions on the future demands as well.

6.5 Results

In this section, we present both objective subjective results, based on 16 subjects, corresponding to 8 groups, where each group included one driver and one helper.

6.5.1 Objective Measures

Mean task completion time and standard deviation per condition and task are presented in Table 1; all data are given in minutes rounded to two decimals. Since the number of subjects was limited, we could not gather sufficient data to prove a significant difference on the time performance between the two conditions tested. According to mean completion time, SEMarbeta performed better on both tasks. However, as mentioned before, the variance of the means was quite high. It is partly caused by the variability in repair skill across the subjects and the unsteady performance of our system affected by the test environment. While no significant differences can be presented, we still observed interesting trends in the data.

Table 1. Task completion time: Means and standard deviations by condition (column) and task (row). All values given in minutes rounded to two decimals.

	VoIP		SEMarbeta	
	mean	***sd***	***mean***	***sd***
Engine task	10.75	3.53	10.42	3.67
Fuse task	8.22	4.13	6.95	4.21

Two important environment factors that influenced our user study are: network quality and lighting conditions. As the user study took place in a basement garage, the wireless network signal could easily be affected or blocked by other facilities. The garage lighting was not bright enough for the camera of the tablet PC to capture clear images. In one round of the user study, a subject could not see the switch of the engine hood clearly, which prolonged the completion time of that round severely and pulled up the average completion time of SEMarbeta system to some degree. However, this also showed the limitations of our system, which cannot be used at night or under poor lighting conditions.

6.5.2 Subjective Measures

After each group of two subjects, the subjects were interviewed on system usability. Some questions were tailored to driver subjects or helper subjects, and some were common for both kinds of subject. Table 2 presents all the questions and some representative answers.

Table 2. Subjective results: All questions (left) and some representative answers (right).

Driver subject questions:	**Driver subject answers:**
1. Do you think SEMarbeta can provide better quality of help compared with voice call?	• Users may have different backgrounds, different languages and different abilities to fix the car. The system can help people in such situations. • It performed badly in the dark environment, but others are good, which can easily show something to the other side.
2. What do you suggest on the sketch function?	• It is easy to use, but it is difficult to indicate the position if the worker is moving. • It is a little complex to select the eraser function, and only one color is available.
3. What do you suggest on the gesture function?	• I could see the helper's instruction with his finger pointing things out. • I saw the gesture from other side, but it was too big. It could be more helpful if the image was smaller.
4. Would you repair your car by yourself if you have SEMarbeta system?	• I would like to use it if the problem is more complicated, such as electronic repairing. I can solve them according to instructions, rather than read manuals.
5. Which device would you prefer to run this system? Phone or Tablet PC?	• Technically, having a tablet is better for the bigger screen to see the instructions, but I will not buy a tablet just because of having this system in my car.
Helper subject questions:	**Helper subject answers:**
6. Have you helped your friend on fixing things via phone before?	• I used to help them on computer problems by phone call. It was hard to understand what the other side was doing.
7. Do you think SEMarbeta can provide better quality of help compared with voice call?	• Yes, it enables users to see more. Sometimes it is difficult to describe a thing with only oral explanations, since people always don't know how to call it. With this kind of system, I can point it out and verify the other's actions.
8. What do you suggest on the sketch function?	• I could paint something on the screen to show what I mean to the other side. • It should have some transparent properties. Because sometimes the sketches from two sides may overlap.
9. What do you suggest on the gesture function?	• It is easier to point out than paint with fingers or the cursor. However, sometimes I forgot this function. It may take time to learn and accept it. The Interaction way is good and intuitive. • Since it is the first time to use it, people may be confused by the hand showing on screen and consider it belongs to the other side.
10. Would you try to help your friends on fixing things if you have SEMarbeta system?	• If I know how to fix the things, I will use the system to fix the problem. For example, I can fix computer programs, point out the button that my father need to click and so on. Video streaming is a good feedback for instructors while helping others who do not have enough knowledge to fix the problem.

Besides the positive feedback, other interviews raised some problems related to the system performance, the GUI of the SEMarbeta application, and the interaction using gesture input. As expected, subjects expressed their positive attitude towards having SEMarbeta as a remote support solution to take the place of the former telephone solution (100% positive). However, from the observation and the final interviews, it also turned out that subjects using the system the first time had some recognition problems on the additional overlays.

7. DISCUSSION AND FUTURE WORK

Only 25% of driver subjects used the sketch overlay to outline objects on their tablet PC. There are three main reasons for ignoring this function. The most frequently given reason was that it is more comfortable to point to objects directly with the finger instead of outlining things on the screen. Another reason was that the outlining would not work when moving the tablet PC, and thus users wanted to have predefined drag-and-drop shapes to highlight things more easily.

On the helper side, there was a big difference in the usage of the sketch overlay and gesture overlay. 87.5% of the helper subjects (or: helpers) used the sketch overlay to outline things for the driver. In contrast, only 25% of helpers provided gesture instructions to the drivers in a sufficient quality. Subjects expressed two main reasons for this. Three helpers did not remember this function although a short introduction of this functionality was given prior to the test. Two helpers thought the video communication and sketch overlay were sufficient to solve the problem.

Since only a few subjects used the gesture function, we discussed possible reasons for this. Most helpers stated that the sketching function is a more common and natural way of interaction than using gestures. Some of the helpers focused on looking at the touch-screen and sketching on it, rather than moving their right hand to the black pad and providing gestures. This kind of inconsistency of interactive actions may have reduced the cognitive support provided by the system.

It is possible that the gesture capturing techniques used in CollaBoard may perform better than the solution we currently

employed. This is mainly because CollaBoard uses pointing at the position where the user also sees the object, while SEMarbeta requires indirect pointing, so that input and output are not collocated, but coupled through the captured image of the finger. This indirect pointing could be avoided if a camera would capture the helper's interaction directly on the screen. However, as we mentioned in previous sections, the required segmentation for this has some limitations, and this kind of interaction may cause fatigue for professional helpers who have to work long hours.

In future work, we plan to improve information presentation, which has the end of reducing cognitive load. Current gesture presentation is still quite limited by the segmentation algorithm and image processing ability of mobile devices. We believe the interaction concept of Kinect can be a way to realize gesture recognition in front of the screen. The concept of capturing body movement to control the movement of a 3D skeleton and then rendering the skeleton to a 3D avatar could bring two benefits to our system. From the user study, we learned that helpers always make gestures during their communication, even though they knew their gestures could not be captured by our system. It is a natural response when helpers meet some difficulties in speaking out the exact word they want to describe the operation. If a device similar to Kinect would be used in our system, the helpers could provide their gestures more naturally during the support process, with no limitations from the system on providing gestures in some fixed area. Moreover, when using a Kinect instead of our image capturing process, gestural input may be less affected by ambient light conditions or background images. Haptic cues, for instance using actuated faders [15], could enable eyes-free control [20].

8. ACKNOWLEDGEMENTS

We are highly grateful for funding, work environment, and advise on user testing offered by the Human Factors group of Semcon Caran AB. This help enabled us to implement our idea in the automotive domain and provided valuable supervision. We also thank members of the CollaBoard team for their support. Morten Fjeld thanks the Swedish Foundation for Strategic Research for enabling him to work at the Innovation Center Virtual Reality (ICVR) at INSPIRE AG, Zurich (SSF grant: SM11-0009).

9. REFERENCES

[1] Alem, L., Tecchia, F. Huang, W.: Remote Tele-assistance System for Maintenance Operators in Mines, In: 11th Underground Coal Operators' Conference, 2011, 171-177.

[2] Anastassova, M. Burkhardt, J.M.: Automotive technicians' training as a community-of-practice: Implications for the design of an augmented reality teaching aid. In Journal of Applied Ergonomics. 2009 Jul; 40(4), 713-21.

[3] Cohen, M., Fernando, O.N.N.: Awareware: Narrowcasting Attributes for Selective Attention, Privacy, and Multipresence. In P. Markopoulos, B. de Ruyter, and W. Mackay, editors, Awareness Systems: Advances in Theory, Methodology and Design, Human- Computer Interaction Series, ch. 11, 259-289, Springer, 2009.

[4] Fussell, S.R., Setlock, L.D., Yang, J., Ou, J., Mauer, E., Kramer. A.D.I. Gestures over video streams to support remote collaboration on physical tasks. Hum.-Comput. Interact. 19-3, 2004, 273-309.

[5] Herskovic, V., Ochoa, S. F., Pino, J. A.: Modeling Groupware for Mobile Collaborative Work. In Proc. CSCW 2009, 384-389.

[6] Kunz, A., Nescher, T., Küchler, M.: CollaBoard: A Novel Interactive Electronic Whiteboard for Remote Collaboration with People on Content. In Proc. CW 2010, 430-437.

[7] Kurata, T., Sakata, N., Kourogi, M., Kuzuoka, H., Billinghurst, M.: Remote collaboration using a shoulder-worn active camera/laser. In Proc. ISWC, 2004, 62–69.

[8] Kuzuoka, H.: Spatial workpace collaboration: A SharedView video support system for remote collaboration capability. In Proc. CHI 92, 1992, 533–540.

[9] Kuzuoka, H., Kosuge, T., Tanaka, M.: GestureCam: a video communication system for sympathetic remote collaboration. In Proc. ACM CSCW '94, 1994, 35-43.

[10] Ladikos, A., Benhimane, S., Appel, M., Navab, N.: Model-free markerless tracking for remote support in unknown environments. In Proc..VISAPP, 2008, 627-630.

[11] O'Hara, K., Black, A. Lipson, M.: Everyday Practices with Mobile Video Telephony. In Proc. CHI 2006, 871-880.

[12] Papadopoulos, C.: Improving Awareness in Mobile CSCW. In IEEE Trans. on Mobile Computing, Vol. 5.10, 1331-1346.

[13] Ran, Y., Wang, Z., Zhu, F.: Trends of mixed reality aided industrial design applications. In Proc. ESEP 11, 3144-3151.

[14] Sakata, N., Kurata, T., Kuzuoka. H.: Visual assist with a laser pointer and wearable display for remote collaboration. In CollabTech, 2006, 66–71.

[15] Shahrokni, A., Jenaro, J., Gustafsson, T., Vinnberg, A., Sandsjö, J., Fjeld, M. (2006): One-dimensional force feedback slider: going from an analogue to a digital platform. In Proc. NordiCHI '06. 453-456.

[16] Spikol, D., Milrad, M., Maldonado, H., Pea, R.: Integrating Co-Design Practices into the Development of Mobile Science Collaboratories. In Proc. 9th IEEE ICALT, 2009, 393-397.

[17] Tang, A., Neustaedter, C., Greenberg, S.: VideoArms: embodiments for mixed presence groupware. In People and Computers XX—Engage, 2007, 85–102.

[18] U.S.News & World Report: Google Glass Unlikely to Be Game Changer in 2013; accessed 14.01.2013 http://www.usnews.com/news/articles/2013/01/02/google-glass-unlikely-to-be-game-changer-in-2013

[19] Villacorta, P. J.: Sensitivity analysis in the scenario method: A multi-objective approach; 11th Int. Conf. on Intelligent Systems Design and Applications (ISDA), 2011, 867-872.

[20] Yi, B., Cao, X., Fjeld, M., Zhao, S. (2012): Exploring user motivations for eyes-free interaction on mobile devices. In Proc. CHI '12, 2789-2792.

[21] http://www.arvika.de/www/e/home/home.htm; accessed 14.01.2013

[22] https://plus.google.com/111626127367496192147#111626127367496192147/posts; accessed 14.01.2013

[23] http://www.samsung.com/ch/consumer/mobile-phone/tablets/tablets/GT-P7500UWDITV; accessed 14.01.2013

[24] http://research.microsoft.com/en-us/projects/conferencexp/; accessed 14.01.2013

[25] http://en.wikipedia.org/wiki/User_Datagram_Protocol; accessed 14.01.2013

[26] http://en.wikipedia.org/wiki/Voice_over_IP; accessed 14.01.2013