Rethinking the Interactions Between People and Cars

Casper Kessels
2018

Master's Thesis

Rethinking the Interactions Between People and Cars

Author
Casper Kessels

Supervisors
James Eagan
Télécom ParisTech

Marc Pinel
Groupe Renault

Hosting enterprise
Groupe Renault

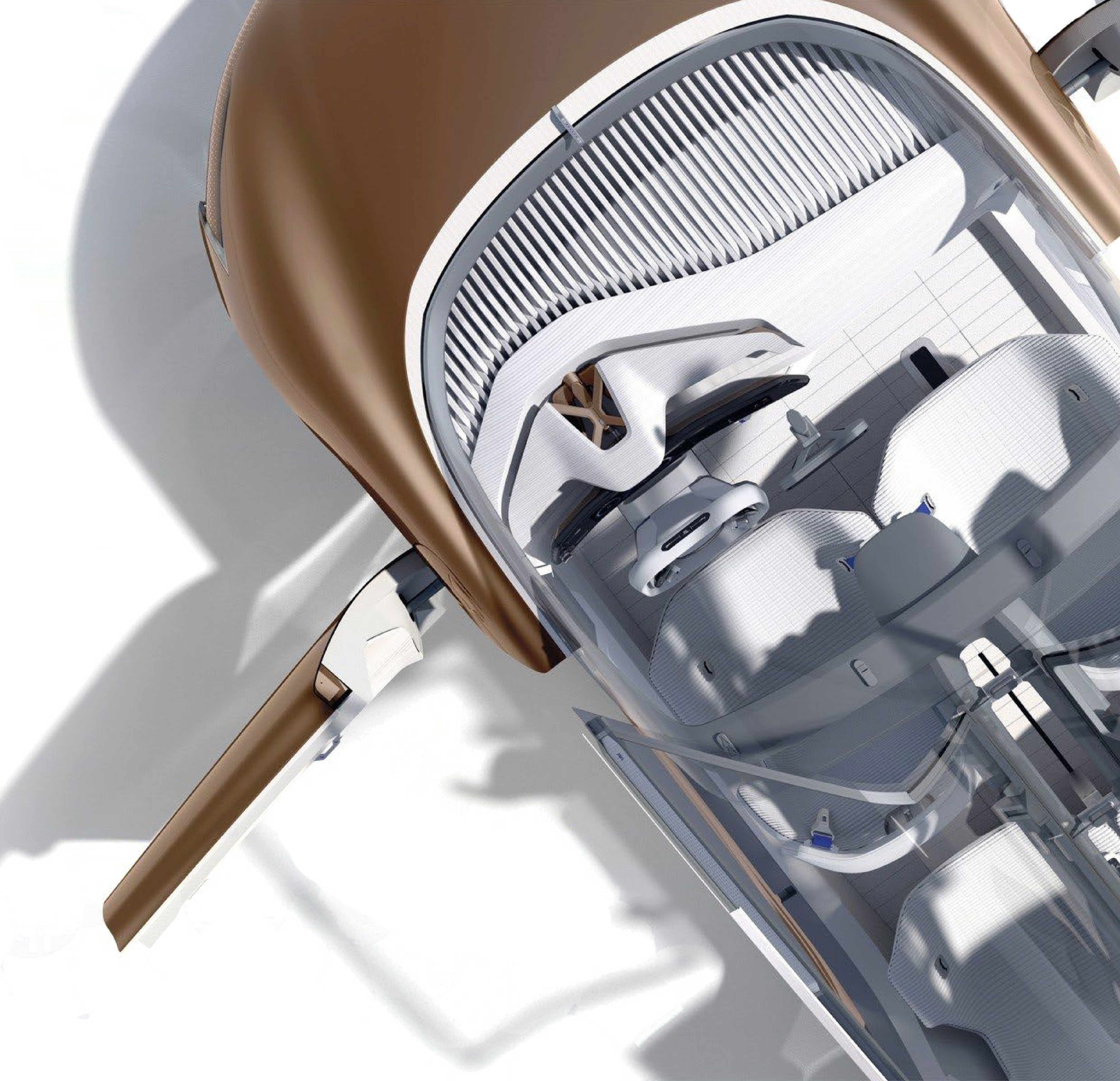
15/03/2018 - 15/09/2018

Master in Computer Science - Human Computer Interaction

Université Paris-Saclay
Secrétariat:
tel: 01 69 15 66 36
fax: 01 69 15 42 72
email: murielle.bernard@u-psud.fr

Summary

This project presents a new way of providing interactions between people and the car. The concept consists of three different parts. First a new way of interacting between the driver and the cluster using gesture interaction. The second part describes why and how the car can be designed around the smartphone. The last part focuses on the feedback from the car and how this can be made more natural. A user test was conducted to test the discoverability of the gesture interaction on the steering wheel. Particularly, the user test explored the use of touchpads on the steering wheel, a totally new application in the automotive industry. 14 participants were asked to perform several tasks using a low-fidelity prototype. The test results highlighted a high discoverability potential of gesture interaction and two main points of improvement to ease the process for users.



Glossary

Cluster	Screen behind steering wheel with information like speed and RPM
Multimedia screen	Screen often placed in center console with media and navigation
UX	User experience
UI	User interface
ADAS	Advanced Driver-Assistance Systems
AI	Artificial Intelligence

01

Introduction

1.1 Introduction

Ever since the invention of the first car with an internal combustion engine, the development of the car and the interaction between people and cars have been changing. Pre-war era cars are notorious for being incredibly complicated to control, with buttons, handles and pedals distributed all over the interior, and exterior, of the car. Controls also varied from manufacturer to manufacturer, and even from model to model. This made driving a car incredibly complex. It took until the 1920's for reasonably accessible basic controls to be implemented in the car.

The basic interaction between vehicles and humans, the layout of the steering wheel, pedals, and gear lever, have stayed roughly the same ever since. However, more and more functionalities have been added to the interior of the car with each new generation. For example climate controls, radio, lights, seat adjustment, etc. In the past 20 years, the car has seen the beginning of a new revolution. The capabilities of computers have grown exponentially. Users like to see more and more technologies in their cars, starting with CD players in the 80's to conversational agents, touch screens, and self-driving technologies today. But the capabilities of people have not grown as fast as the capabilities of the computer and all of these functionalities need to be controlled by the people in the car. With each new feature, the car becomes a little bit more complex. This, in combination with new kinds of interaction, such as touch screens, lead to car manufacturers creating separate user experience design departments. It is the task of the designers working in these departments to bridge the gap between the high capabilities of the technology and the capabilities of people and to make the interaction between users and cars as simple and fluid as possible (figure 1).

Cars are incredibly complex products. It takes around 4 years to develop a car. And in these 4 years, countless departments from various disciplines have to work together. Design, engineering, marketing, product planning, legal, finance, communication, these are just some of the departments that are involved, each consisting of several specialized subdepartments. A UX design department has to communicate with interior designers, ergonomics, product planning, marketing, programmers, engineers, etc. And each has their own requirements and limitations for the products. Changing the user experience of the car can thus be a difficult process since all of these departments have to be consulted and informed.

Up until today, the interaction between people and cars has been evolutionary. With each new generation of cars, the interactions have been made more modern instead of completely redesigned. This means that all of the new technologies in cars are simply being fitted into the old interaction models. For instance, cars today have a center screen which gives drivers access to information of the car. Each new feature is simply added to this screen. 20 years ago, this center screen only had to display the media settings, but today it displays the media, navigation, apps, settings, etc. This contributed to cars becoming more complex and difficult to operate. But thanks to the rapid advancement of technology, there are a lot more opportunities to design interactions differently than with a touch screen or buttons. There is an opportunity to look at the interactions people have with cars and to design them from scratch.

This report describes a project that was done for Groupe Renault. The main goal was to investigate new ways of designing the interactions between people and cars. The research consisted of three parts: an exploratory phase, where the challenges of challenges of interactions and cars were assessed; a prototyping phase, where the prototype was designed; and an evaluation phase where a solution is proposed, tested and evaluated.

1.2 Renault

1.2.1 The Company

Founded in 1898, Renault is one of the oldest car brands currently in existence. Today, Renault, officially named Groupe Renault, has grown into a group consisting of Renault, Dacia, Renault-Samsung Motors, AvtoVAZ, and Alpine. The group also has a division called Renault Sport which is responsible for creating sports versions of road cars, and several racing endeavors like the Renault Formula 1 and Formula E teams.

The group has an alliance with the Nissan Motor Corporation called the Renault-Nissan-Mitsubishi Alliance, after Nissan acquired a controlling interest in Mitsubishi in 2016. In 2017, the alliance was the third best selling automotive group after the Volkswagen group and Toyota, with 10.07 million vehicles sold [6]. Renault was the 9th best selling brand in 2017 worldwide, and 2nd in Europe with 2.6 million and 1.2 million vehicles sold respectively [14].

The Groupe Renault is active in all continents except North America. European sales count for about half of all sales globally.

As a brand, Renault is positioned as a 'peoples car', providing high-volume transportation to the masses. Renault is seen as an iconic French brand with a strong legacy thanks to high sales success with the Renault 4, 5, Twingo and Clio. The company wants to be an accessible brand that is close to the people and creates products that are loved and contain a certain 'joie de vivre'. This is all reflected in its slogan: 'passion for life'.

Today, Renault is active in almost every segment of the car market, from small city cars to pick-ups and vans. In the past 5 years, sales numbers have increased greatly. Renault has formed successful partnerships and tries to innovate and

Image 1. The Renault line-up as of 2017. From left to right: Zoe, Clio, Mégane, Scenic, Talisman, Espace, Koleos, Alaskan, Kadjar, Captur and Twingo.

to explore new markets to enter. A big target of innovation is electrification which has resulted in the development of Zoe, a small, electric city car, and at least 10 other electric models to be released in the coming years.

1.2.2 The Design Department

Groupe Renault's main design department is located in the Technocentre in Guyancourt, which is home to more than 13,000 employees. In total, there are more than 500 designers active around the world, with over 400 working in the Technocentre.

The design process of a car is long and complicated so there are many different fields involved like exterior design, interior design, clay modeling, 3d modeling, ergonomics and UX design.

The UX department is responsible for developing the user experience which includes interface design and interaction design. Initially, the work of the department was small and included just the cluster and center screen. Thanks to the increase of technology in the car, like touch screens and head-up displays, the department is quickly growing and its scope is expanding to include any interaction between a person and technology in and around the car.

In the next years, the department will grow even more in size to cope with the increase of technology in the car. Also, more work will be focused on exploring and developing visions and concepts for future ways of interaction. However, today, the UX department does not have enough resources to develop new concepts of interaction. Therefore, this project was commissioned.

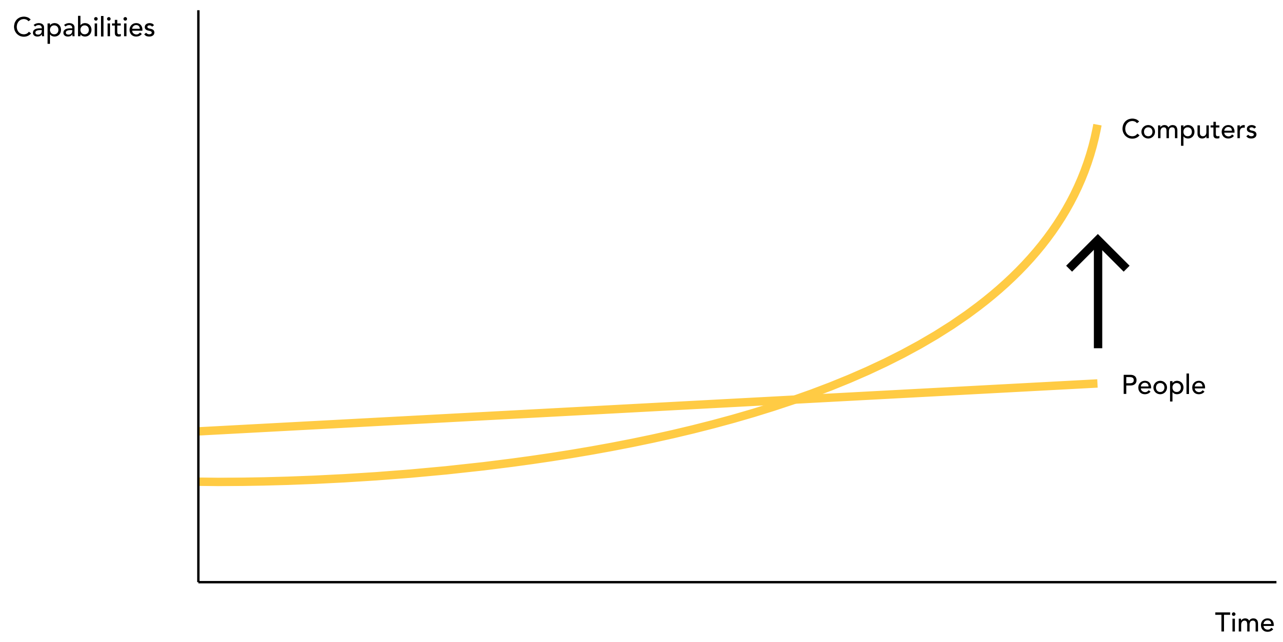
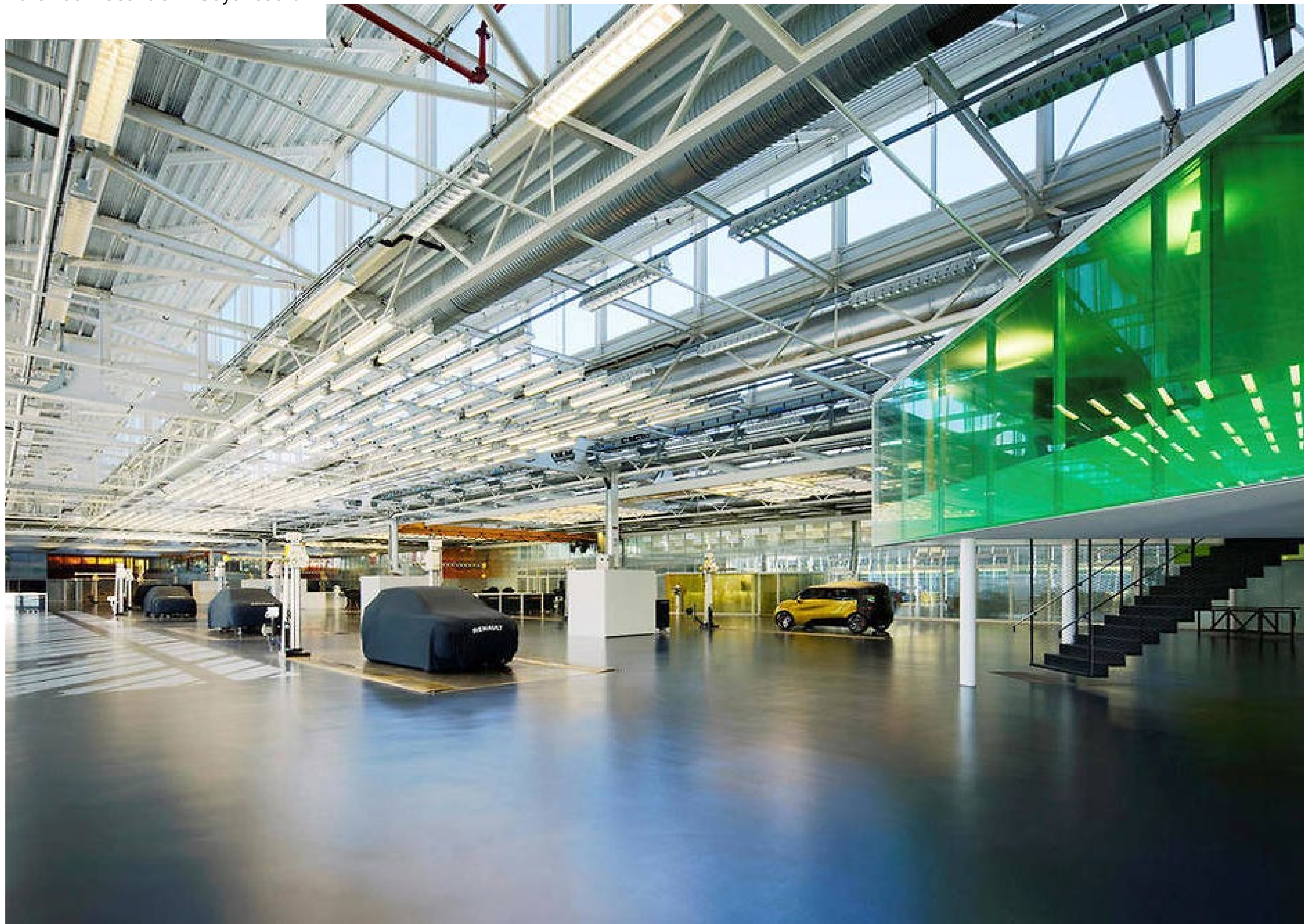


Figure 1. Computer capabilities compared to human capabilities [3]


The Renault Design Department in the Technocentre in Guyancourt

02

Concept

2.1 Problem

Technology is rapidly changing the car. 20 years ago, the most advanced feature on a Mercedes S-Class, widely seen as one of the most innovative production vehicles, was parking sensors. Today, the Mercedes S-Class can park itself.

On the other hand, the interaction between a person and a car has not changed much at all. Drivers still use traditional keys, most of the controls are still located in the same place, the cluster shows the same basic information, etc. These interactions have become more modern though; keys are now full of sensors, touch screens are replacing buttons, clusters are now completely digital, etc. So there has been progress but more in an evolutionary way, rather than a revolutionary way.

Naturally, one should not change just because it is possible, but rather because it is necessary. Driving a car is a dangerous activity thus drastically changing the user experience can lead to accidents and even deaths. Sticking to traditional and familiar controls does not confuse drivers. However, as more features are fitted in cars within the traditional interaction models, they become increasingly complicated to operate. Research has shown that most people are not aware or are not using the technologies in their cars. And that in many cases, users prefer to use their own smartphone and tablets because they are familiar with them and they work well [9]. Combine this with more external distractions, like the smartphones, and the result is that today, technology in the car can cause a lot of distraction. A recent study from Cambridge Mobile Telematics showed that phone distraction occurred during 52 percent of trips that resulted in a crash [17].

As described above, Renault is still expanding its UX department. At the moment, designers are busy keeping up with the development of systems of the cluster and multimedia screen. Therefore, not much focus has been given to the exploration of different interaction systems. That is where this project comes in. In this report, a concept is presented which does not look at the current interaction systems of Renault but instead, the question was asked: if it was possible to design the interactions between a person and a car from scratch today, how could they be designed?

2.2 Scope

With such a broad question, it is important to define a clear scope. First, the requirement is that the design should hypothetically be released within 3 years, so using any technology that is not ready before 2021 is not possible. Second, the concept will stay away as much as possible from the topic of artificial intelligence and self-driving cars. This has multiple reasons. It is simply not possible to predict how fast the development will go in the field of artificial intelligence so it makes no sense to focus on it. Also, there is an incentive that when encountering a difficult design problem, to let AI take care of it. In the mind of the designer, AI agents are often flawless and perfect systems but in reality, the opposite is true.

On the other hand, there are no other restrictions. So the cost of the technology is not relevant to this concept. Also, any software constraints are mostly ignored. This means that when an idea is technologically possible but limited due to available software products, this constraint is ignored and the best possible solution is assumed.

2.3 State-of-the-Art

Today, the technology in the interior of Renault passenger vehicles vary based on the type of car. Cheaper models like the Renault Twingo have a more basic interior than expensive models like the Espace. Also, per model, different interior options are possible.

R-Link is the name of the digital system that Renault designed. This includes the navigation, multimedia, and more. It is available in every car, except the base version of the cheapest models. Depending on the model and interior option, R-Link comes in two different versions, 1 and 2 (depending on when the car was released), and on two different screen options, a 7-inch horizontal screen, or a 8,3-inch vertical screen.

The most expensive interior option, available in the higher-end models of Renault, features a 8,3 inch vertical touch screen in the center console (image 2). With R-Link 2, the latest version of the system, users can access all of the functionalities of the car via the screen, except for the climate controls and driving mode. Though, specific settings for the climate control and driving mode can be changed via the touch screen.

The cluster consists of a digital display with the main information like speed and rpm, and one analog display (LED strip) on either side of the

Image 2. Interior of the 2016 Renault Talisman Initiale

digital display which show the fuel and water temperature. The driver can interact with the information on the cluster via buttons on the steering wheel. Also, the layout of the digital display changes depending on the driving mode.

There are more buttons on the steering wheel and they control basic much-used functionalities like answering a phone call and changing the volume.

2.4 The Concept

The concept completely rethinks all the interactions between a person and a car. Consequently, almost all of the interactions are different. As mentioned before, today, the interactions of the car are based on evolutionary design. This concept, however, looks at the interaction from a new perspective, with the technology of today. And with a focus on the most important problems facing the interaction today, such as driver distraction.

The concept consists of three main parts. First, the critical interactions between a driver and the car while driving are moved from the central screen to the cluster. These are for instance the media controls and navigation controls. Second, the concept does not feature a center screen with the main access to the computer. Instead, this is moved to an online environment that can be accessed via a smartphone application or website. Last, there are interactions like the climate controls and volume controls that remain in the car, but as physical buttons instead of on a touch screen. Also, the feedback from the car, like warning messages and other ADAS displays are made to be more natural. What all of this means explicitly is explained below.

2.3.1 Cluster

Today, almost every car has a cluster screen which has basic controls on the steering wheel, and a center screen which is often a touch screen. One can look at this as if there is a computer in the car, that can be accessed by the people in the car, which has two screens with information: the cluster shows essential information to the driver, like the speed and the current media, that can be accessed via buttons on the steering wheel. And the center screen which is the main access to the computer of the car. This closely resembles a tablet, both in hardware and software, which shows all the information that can be useful to the driver and the passengers.

The need for a computer in the car is obvious. The driver needs to navigate, play music and change the settings of his car. The current interaction with the computer, however, is not obvious. The use of a touch screen is very attractive, gives a lot of flexibility and gives the car a futuristic look, but it does not improve the usability while driving as drivers always have to look where they have to press instead of blindly reaching for a physical button or knob. Also, operating systems are difficult to learn. Most people only use 1 or 2 operating systems in their daily life, one for their computer and one for their smartphone or tablet. The system in their cars, like R-Link for Renault, is a totally new operating system that they have to learn. To make it easier to use the system, designers try to make them as close as possible to existing systems, like iOS and Android. But they can never exactly replicate these designs.

As a result, people struggle to use the features of the system or they ignore it and use their smartphones or tablets instead.

This concept proposes a new solution which is to show the essential information on the cluster and provide an easier way to interact with the information there. And to move the complicated interactions to the systems that people are used to, like their smartphones.

In the cluster, to reduce visual distraction, the information should be presented as close to the field of view of the driver as possible. Therefore, most information will be displayed on the cluster screen. This raises two problems: what information should be shown as, currently, the center screens shows as much information as possible, and how to manipulate the information, since it is not possible to touch the screen. The cluster will only show the information that the driver needs while driving, all the other information is moved away to other parts of the system. It is important that the cluster is as simple as possible so it will only show the media, navigation, ADAS information, and essential phone notifications.

Still, that leaves the problem of how to interact with this information. Today, the cluster is controlled via buttons on the steering wheel. A similar solution would be ideal because the driver would not have to move his hands to operate the cluster. Simple, physical buttons do not provide a lot of flexibility and there is a certain disconnection from the screen when using buttons. Thanks to the presence of touchscreens in our lives, people have become used to directly touch on interactive elements and manipulate them with different gestures. The ideal solution has the location of the buttons on the steering wheel and the manipulation of touch screens.

Research in the field of human-computer interaction focused on automotive applications has been growing in the past years. Multiple researches have shown promising results of new types of interaction that allow more direct manipulation and less distraction [13, 7, 4, 1]. Most of the research focuses on a type of interaction that allows the driver to keep his hands on the steering wheel, either via gestures or via voice command. The work of Döring et al shows promising results in having gesture interaction on the steering wheel [4]. Together with research on multi-modal interaction [13], this will form the basis of the interaction of the cluster. With gesture interaction on the steer-

Image 3. Gesture interaction
Top: Cluster layout with the two touchpads
Bottom-left: Graphical representation of swipe gesture with right touchpad
Bottom-center: Graphical representation of swipe gesture with left touchpad
Bottom-right: Custom gesture visualized

ing wheel, combined with speech interaction, the concept will fit the requirement of having the right location on the steering wheel and similar manipulation to touch screen.

The final concept has two touchpads on the steering wheel close to the thumbs of the driver. The driver can operate the cluster through the gestures that he is used to from his smartphone and tablet. These gestures are swiping, pinching, and tapping. Next to these gestures, the system will also enable the possibility to set up completely customizable gestures.

The concept has a touchpad on the left side and one on the right side of the steering wheel. The left touchpad controls the main programs of the computer. It is used for switching between media, navigation, ADAS, and phone. The right touchpad controls the submenu of each program. To navigate through the menu's, the user can swipe in 4 directions, up, down, left, and right, and each direction corresponds to a setting or function.

The touchpads can also be used together, at the same time, to simulate the same gestures that are used on smartphones and tablets like pinching and scrolling.

Another great advantage of using touchpads is that custom gestures can be used. The custom gestures allow users to define their own gesture for a specific interaction they often do. For instance, when a user texts his partner every time he leaves work to go home, he can set up a custom gesture so that with one gesture, he can do an action that would otherwise take 5 actions or more. This custom gesture can be a letter, or a symbol like a heart.

With custom gestures, users can also, for instance, use handwriting to fill in destinations of the navigation system.

In the current design, there are essential gestures, like the menu navigation, and basic custom gestures like a 'check' and 'x' gesture for approving and canceling respectively.

But since users can set up their own custom gestures and choose exactly what these gestures control, they can appropriate the system to their own needs and uses. An expert user can choose to set up as many gestures as he wants. But by default, the system only has the bare minimum of gestures to keep it simple.

The steering wheel is not a static object. Actually, it is the least static object in the interior. While driving, the hands of the driver will move often so there won't be perfect conditions for the input of the gestures. Luckily, the gesture input can be very forgiving. These fluctuations of movement can be taken into account while designing the system.

The gesture interaction allows a user to keep his eyes on the road while interacting with the system. Consequently, the user will not always see direct feedback of his actions. It is important that the user is confident in using the system and that he feels that he is in control. In the case that a gesture is misinterpreted by the system, there is an 'undo' gesture that a user can do to undo the previous action. This gesture can be reused for every action. It is there to give the user confidence in using the system so that whatever he does, can be undone with a basic gesture.

Next to the gestures, the driver will be able to execute voice commands. This will add redundancy to the system so the user can choose which kind of interaction he prefers for each functionality. Today, it is already possible to interact with the car via basic voice commands. This interaction is not perfect right now but the technological progress in this field is promising. Especially since big players with the best voice agents, like Google and Amazon, have started offering their software to third parties.

The goal of the voice command is to give the driver an extra possibility to interact with the system. The voice command works especially well when executing complicated interactions like filling in a destination in the navigation system.

Discoverability

The main problem with using gestures is discoverability. How can users discover how to use the system if the interface is hidden? To solve this problem, the system detects when a user is hesitant and shows an interface that teaches the user how to operate the system. For instance, when the user wants to change from media to navigation, he has to swipe up on the left touchpad. But when the user does not know the gesture, he might hesitate for half a second, at that point the interface shows a graphical representation of the interaction on the cluster to help the user out. The next time, the user knows that he has to swipe up to change to navigation. If not, he can touch and hold again on the touchpad, and the interface will be shown again. For each gesture interaction, there is a graphical representation that teaches the user. The study hypothesised that like this, users learns all crucial gestures over time so the feedback will not be necessary most of the time.

This idea works well for the menus since they have a clear structure by swiping in four different directions, but for the custom gestures, this will not work. Bau & Mackay (2008) present an interesting solution to this problem [2]. If a user starts a custom gesture, but does not know how to finish it, the same principle applies as before. After hesitating for half a second, the system shows how to finish the gesture with a graphical representation of the gesture on the cluster. The system will show which gestures the user can execute from the point where he got stuck.

2.1.2 Designing around the Smartphone

Today, most services are available and consumed on a smartphone. News, sports, shopping, calendar, photography, etc. are all most used on smartphones. These apps have all used common principles to design their interfaces. Hence, when using an app for the first time, a user does not take long to master the interface.

However, in a car, users face new systems, not similar enough to the smartphone systems they are used to. This leads to much confusion and distraction [16]. Also, one of the effects is that most people are not aware of all the features in their car [9].

Apple and Google have realized this problem and have released their own systems for the car, Apple CarPlay and Android Auto respectively. Both systems extend the screen of the phone to the screen in the car and offer certain apps to the user. The popularity of these systems highlights the problem described above [18].

Also, people use these systems because they come with a very powerful app landscape. These systems allow users to use their favorite media and navigation apps that are often better than the systems that car companies provide.

Some users who don't have CarPlay or Android Auto or don't want to use these systems, simply use a mount to attach their smartphone on the dashboard of the car [9]. People clearly prefer to use their smartphones over the systems of car companies. So instead of trying to create better in-car systems, why not design around the use of the smartphone?

How it works

By designing around the smartphone, the aim of the concept is to provide users with a familiar interface in combination with the app landscape and connectivity that they are used to. The computer of the car can be accessed via an online environment, like a smartphone application or website. In this application, the user can interact with the entire system of the car. Today, users have to be inside the car and interact with the multimedia screen to have access to all the features of their car. Although some car manufacturers offer connectivity via smartphone applications, what users can do with them from a distance is limited.

However, this concept places the smartphone as the main control of the car's computer.

One of the key points of this system is to move complicated interactions away from the multimedia screen of the car and to the smartphone, like setting up a route on the navigation system and changing settings of the car. The idea behind this is that once someone enters the car, the focus is on going from A to B and not on discovering all the features and settings the car has to offer. But when these items can be accessed from the smartphone, users can play and discover the settings of the car whenever they want, it is not necessary to enter the car for that. For instance when they have spare time at home or when they are bored while waiting for the dentist.

As reducing driver distraction is the main point of this project; while driving it will be impossible for the driver to access the online environment. For crucial interactions, like setting up a route, it is possible to use the cluster. But to change the settings of the car the driver has to stop and use the smartphone. The reason why this was chosen is to force drivers to stay off their phone. Changing settings of the car is something that is not urgent while driving and therefore, it can be moved out of the car and to the smartphone. It simplifies the interface of the cluster, and forces users to keep their attention to driving.

Not only is it important to design around the smartphone from a software perspective, but in the physical world, it should also be made clear to the user.

For each passenger in the car, including the driver, there is a designated area for placing the smartphone. These areas provide wireless charging and are also the points where the connection between the computer in the car and the smartphone is made. To use the smartphone in the car, an application has to be installed. After that, anyone

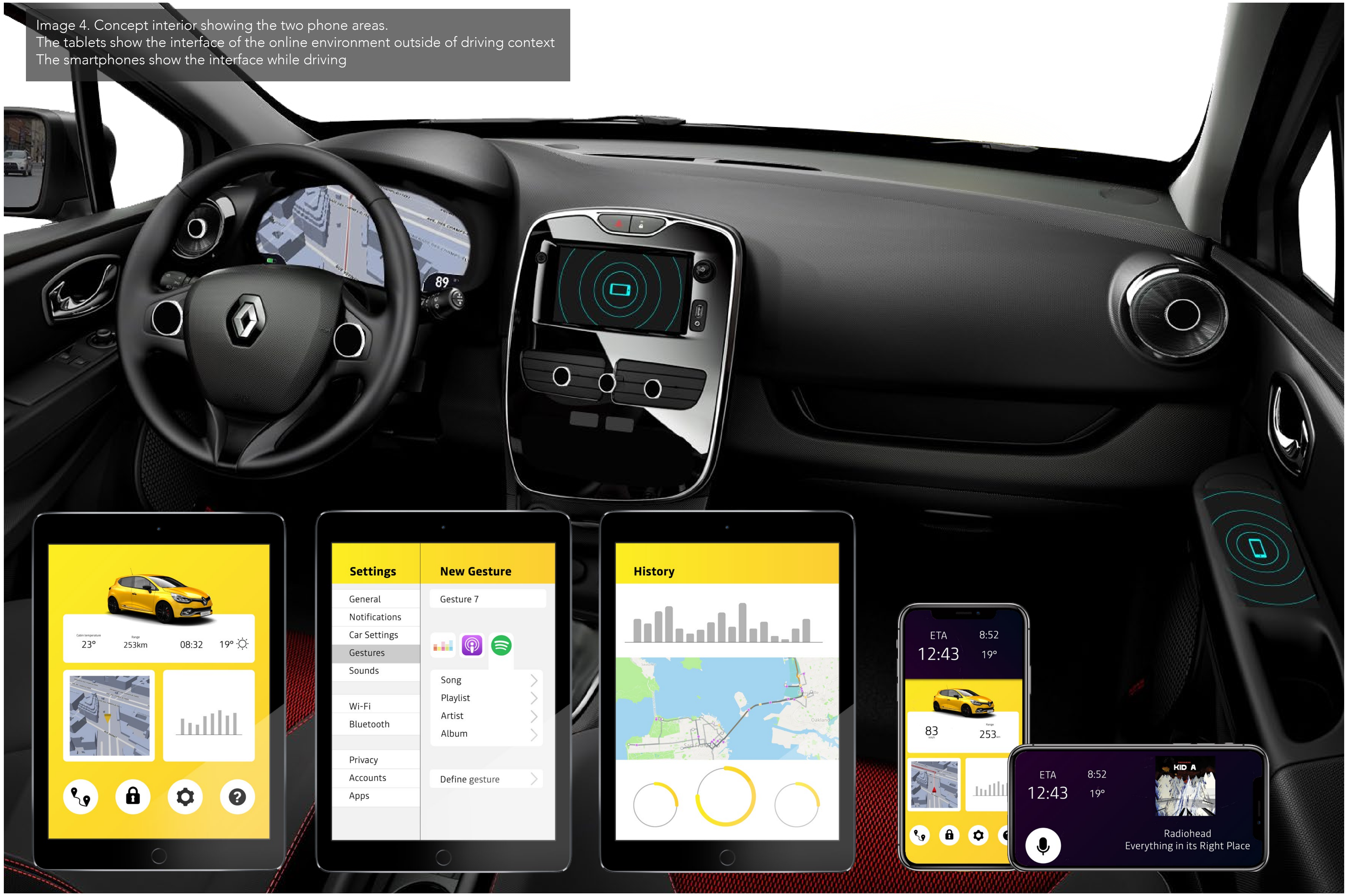


Image 4. Concept interior showing the two phone areas.
The tablets show the interface of the online environment outside of driving context
The smartphones show the interface while driving

with a smartphone and the application can place the phone in the designated area and it will be connected to the car. This means that anyone can interact with the car via his smartphone. So when two people are in the car and the driver is looking for a petrol station, he can ask the passenger to help out and send the updated route to the car. The same goes for media, settings, messages, etc.

Image 4 shows a possible design for the interface. The interface copies the design language of iOS and Android so users need minimal time to adjust to the interface. This application is also where custom gestures can be added.

As explained before, to reduce driver glance, most information is presented on the cluster, right under the focus of the eyes of the driver. This is beneficial to the driver, but not to the passengers who are missing a central display with information about music, navigation, etc. Therefore, there is a main phone connection area in the center console. The difference between the 'main access area' and a regular phone area is that this is the only spot that is in reach of the driver so placing the phone there will restrict the usage of the phone completely to the driver. It will display essential information like the current media that is playing, the time and ETA. The only interaction with the smartphone that is possible is a voice command. There is no other way to use the smartphone when it is placed in the main access area unless it is taken off it. And even when it is taken off the spot, while driving the application will not allow any input from the driver to prevent distraction [10]. This is done to discourage the driver from using his smartphone while driving, even to use other applications and services. Passengers do have the possibility to go into the settings of the car via the smartphone app, but the driver will not be allowed to do this.

In the interior of the car, only a smartphone can be used. But outside of the car, the user can access the online environment with any internet-connected device like a tablet or laptop.

Another feature of the system is that a lot of the sensor data of the car is accessible to the driver from the smartphone. People are more and more interested in acquiring data about themselves and their behavior. Whether it is via smartwatches or their thermostats. After a house, a car is the most expensive purchase people will make. Being able to get an insight into your driving behavior, driving style, costs, and maintenance can be very useful. Today, cars are loaded with all kinds of different sensors and are immense data mines. Users will have access to the cameras of the car from a distance so that they can always see what is going on in and around their car. Also, users get an insight into their driving behavior by seeing when and how they use their car during the week, how much fuel they use, how much kilometers they drive, how hard they accelerate, how hard they brake, how much time they spent in traffic, etc. All of this can be compared to other drivers to see how they can improve their driving to be safer or more economical.

2.1.3 Natural Interaction

The last part of the concept focuses on the interactions that happen outside of the cluster and smartphone. These are for instance the climate controls, but also the feedback the car gives to the driver, for instance from the ADAS systems. To reduce driver distraction, it is important that these interactions lead to a minimum amount of cognitive load on the driver [15, 5, 8]. The idea is to make them more natural so that the driver has more mental resources to focus on driving [11].

When looking to the future, cars will be more and more autonomous. A great metaphor for how this will be, is to look at a rider and a horse. Both the rider and the horse have a brain and make decisions, but the rider is always in control. This can also be said about a car and a driver. However, when a rider directs his horse to walk into an object, the horse does not start beeping at the rider.

In a car, the main way of providing feedback to the driver is via warning lights and beeps. There is not a single example in nature where humans get feedback from beeps and warning lights. A common example is when driving too close to the car in front, the car might show a warning light and drivers will be informed via a beep. At that point, they do not know whether the car beeps at them because of their seatbelt, because the lights are on, because a door is open, etc. The drivers are alerted by an unexpected, unidentifiable beep. They then have to divert their attention to the cluster where a warning light tells them that they are too close to the car in front. They then have to look up again and change their driving behavior. It all seems very unnatural, and as a result, it requires a lot of cognitive load. All of the information is transferred either via visual or audio feedback, or both. Using more modalities can reduce the cognitive load of the driver [12].

The second issue is that the controls for functionalities that a driver uses frequently, like volume and climate control, are often placed in the center console because there is where there is space to put them. Not because it is the most logical location. Even more, today, many car manufacturers are opting to implement these functionalities in a touchscreen. As a result, drivers cannot blindly reach for the controls but always have to divert attention from the road to the screen to see where they press. Placing the input in more natural places and choosing a better type of input will allow drivers to use them with minimal distraction.

The main idea behind the design of these interactions is to reduce the cognitive load on the driver of the interaction by designing them to be more natural. Explicitly what this means is that for each interaction the right modality is chosen for the input and output. And, for the input, the right location has to be chosen.

Feedback

First, let's look at how a car warns a driver. Currently, there are two ways: via a beep or a warning light/icon. Cars are increasingly being equipped with more technology like self-driving features. And all of these systems use beeps to warn the driver. It started with parking sensors, but today there are lane departure warnings, cruise control warnings, blind spot warnings, following distance warnings, etc. When taken out of context, it is impossible to tell what a particular beep signifies.

By looking at how humans in a natural environment are warned for danger, it is possible to reduce the cognitive load in the car [11]. For instance, in nature, humans mainly rely on hearing to localize a moving object outside of their field of view. So the same will be done in the car. For instance, when a cyclist is in the blind spot, the sound of the cyclist is amplified through the car's speaker system and by using multiple speakers, the location can be accurately tracked.

Additionally, by using screens displaying a live camera feed instead of the outside and inside mirrors, a greater field of view is displayed. Also, the driver's eyes are tracked. When the driver has not noticed an object in his mirrors, this object can be highlighted since the mirrors are screens. For instance, when a cyclist is overtaking the car, and the eye tracker detects that the driver hasn't seen the cyclist yet, the screen can draw the attention directly to the subject.

The same is done for objects in front of the driver. A light strip is integrated below the windshield on the dashboard. When the driver has not yet noticed a pedestrian, the area below the location pedestrian within the frame of the windshield softly lights up. This will draw the attention of the driver in a natural way directly to the subject, instead of first to a warning light in the cluster after which the driver has to find the danger himself.

Another example of using the right modality for the feedback is by using haptic technology. In the example of the rider and the horse approaching an object, the closer the horse will get, the more uneasy it will become. In the beginning, it will hesitate, when it gets even closer, it will start to slow down and push back, etc. The car will do the same to the driver when he is parking. If the driver gets to close to the other cars, the car will 'push back' with the gas pedal. Or when getting to close to the car in front while driving, the car will make the steering input a bit lighter and push the pedal back to indicate that the car is not at ease.

Naturally, a balance has to be found between informing the driver about a hazard and instead, making the driver think that his car is broken or unsafe.

By basing these types of feedback on how humans process similar feedback in nature, the cognitive load can be reduced since drivers will not have to spend a lot of energy to think about what the warning message or beep signifies. It might even allow drivers to respond to the feedback subconsciously, similar to how you subconsciously keep track of moving objects in your blind spot while walking through a busy city environment.

Input

The same principles are used when dealing with the input mechanisms. In this concept, information from the car's computer that a driver needs while driving is displayed on the cluster screen. Information that is more detailed and not necessary while driving can be accessed via the smartphone. But there remain some possible inputs that a passenger in the car needs while driving but that is not appropriate for display on the cluster. Either because both driver and passengers need to have access to it, or because gesture input is not the most optimal way to provide access to the functionality.

So these inputs cannot be placed close to the field of view of the driver. Therefore, it is important that the driver can operate these controls while maintaining his focus on the road in front of him. He has to be able to find the controls and operate them blindly.

This requirement automatically demands that the inputs should be physical, much like you see in cars today (except for the high-end cars that use touch screens). However, today, these cars place the controls all together on the center console. These inputs are, for instance, the start/stop button, the climate controls, volume and music controls, seat controls, massage controls, driving mode, etc.

Knobs and buttons that are used often, like music volume, will be operated blindly due to the muscle memory that is acquired over time. Though there do remain some functions that are not used very often and because they are all placed close to each other, the driver will have to look away and see what button he has to press.

What makes this concept different from the cars today, is that the inputs are placed on, or close to, the object that they manipulate: the climate controls are placed on the vents. The music controls are placed on the speakers. The seat and massage controls are placed on the seats. The driver just has to follow the source of the item he wants to manipulate to find the input.

Image 5. Overview of the natural interaction

1. Speakers with play and volume controls
2. Light strip highlighting dangerour objects
3. Main phone area
4. Passenger phone area
5. Screen in the mirror highlighting dangerous objects
6. Climate controls placed on top of air vents
7. Haptic feedback in pedals, seats, and steering wheel
8. 360 degrees audio feedback from objects around the car

03

User Test

3.1 Introduction

The concept is very broad and extensive, therefore, it was decided to focus on only one of the three parts for the user test. For this, the interaction with the cluster was chosen. Gesture interaction is a new concept in the automotive world and it has a lot of potential but also a lot of question marks. The biggest one being the discoverability. Therefore, it was decided to create a prototype of a cluster and steering wheel and test the discoverability of the gesture interaction.

3.2 Prototype 1.0

To conduct the user test, a low fidelity prototype was built. As this is a first, exploratory study of using gestures in an automotive context, building a realistic prototype using a real cockpit, dashboard and steering wheel is beyond the scope. Therefore, the prototype consisted of an iPad Pro showing an interactive webpage that displayed a steering wheel with the two touchpads, and a cluster display. The size of an iPad Pro is quite similar to that of a real steering wheel. Of course, the exact grip of a user's hands holding the iPad is not exactly the same as a real steering wheel but the position is.

The webpage with the interactive mockup was written in JavaScript and jQuery. For the gesture interaction to work exactly as envisioned, it should recognize both swipe gestures, multi-touch gestures and custom gestures (like letters and symbols). However, a recognizer that can handle that does not exist for public use and developing one from scratch was not possible due to time constraints. Therefore it was only possible to opt for an existing gesture recognizer.

For the custom gesture recognition, the 1$ Unistroke Recognizer was used, developed by Wobbrock et al (2017). It provides a simple JavaScript library for adding, recognizing, and removing custom gestures. Even though the 1$ recognizer is very easy to integrate and provides one of the best and lightest recognizers, it is not perfect for this concept because it cannot recognize swipe gestures [19]. Consequently, the swipe gestures were recognized by basic JavaScript event listeners but this also meant that the swipe recognizer and 1$ recognizer could not work at the same time because they would interfere with each other. The prototype was built with this constraint in mind.

The system has 4 main features: navigation, media, car, and phone. The navigation system shows a map which has a submenu with 3 options: navigate to work, home, and the previous destination. The interface shows a fourth option, search, but this has no functionality. When one of the destinations is selected, the interface shows a route on the map. The user can exit this and go back to the empty map by doing a custom delete gesture.

The media screen shows the current media that is playing and has a submenu with 3 options: previous, play/pause, and next. Just as with the navigation screen, there is a fourth option, search, which also has no functionality. The car screen just shows an ADAS screen with basic information but has no other features.

The phone screen shows the notifications from the phone. There is one notification of a missed phone call and one with an unread message. The phone screen has a submenu with 4 options that can be accessed: delete, scroll up, open, and scroll down. Once the missed call notification is opened, the user can hang up the call with a 'caret' gesture, or the user can exit the screen with a 'delete' gesture.

When the user opens the message, the following text is shown: "Do you want to go out tonight? I have two tickets for a concert". The user can exit the message screen via the 'delete' gesture, send a negative reply with the 'caret' gesture, or send a positive reply with the 'check' gesture (image 6). The concept does not allow 'typing' a custom message using handwriting because that was deemed to be too distracting while driving. Instead, the interface shows quick reply options that the user can choose to send via gestures.

Image 6. Cancel, caret, and check gestures

The swipe gestures are used to access the 2 menu's: one for the main features and one for each feature. The custom gestures are activated only users go into a menu option of a function. Swipe gestures are used for navigating through the system. In order to reply to a message, for example, users would have to swipe right to go from the media screen to the phone screen. The custom gestures are used to interact with the functionalities of the system, for instance, to reply to that message.

The menus are hidden by default. The user can swipe and the action will be executed without any feedback. If the user is unsure of which direction to swipe in, by touching and holding for 0,3 seconds or longer, a menu appears on the screen with a representation of the gesture. All the custom gestures always have a graphical representation on the screen.

3.3 Prototype 2.0

The first version of the prototype was shown to 4 UX designers who had previous knowledge about the project but had never used the prototype. They were asked to interact with the system and they were observed while doing so. The results of this preliminary evaluation were used to redesign the system for the user test.

Image 7. Prototype version 2.0
Top: Swipe message
Middle: Tap message
Bottom: Menu design with arrows

One of the main criticisms raised by them was the lack of feedback from the system. When first using the touchpads, the participants did not know whether to tap, swipe, or do custom gestures. All of the participants expected a tap gesture to be recognized but instead, the system recognized each tap as a swipe leading to confusion. Also, whenever they had some form of interaction, the system would register a command and execute it, but there was no way of finding out if it was the correct execution. Only if the pop-up menu was used could the participant check if the executed command was correct.

Another point of feedback was the difficulty to understand the structure of using the left touchpad for the main menu and the right touchpad for the submenu.

This pre-test lead to three changes in the prototype (image 7). First, for each swipe, a notification was designed that popped up in the right top of the display to show which kind of gesture was registered by the system. Second, when tapping one of the touchpads, a message was displayed over the information on the cluster that said: “touch and hold to see menu” for the left touchpad. If the right touchpad was tapped, it displayed “touch and hold to see submenu”. The idea behind this addition is that users could more easily understand that there was a hidden menu and that each touchpad had different functionalities.

Last, the interface for the menus was slightly changed. Instead of having just a circle to represent the touchpad, arrows were added to help the user to understand the interaction of swiping in a certain direction to operate the system.

3.4 Test Setup

The test was focused on discoverability. The participants were given minimum explanations about the system and they were given a short introduction about the project. They were only told two things: they can interact with the system via the two touchpads on the steering wheel, and that the touchpads are just like a touchpad on a laptop, they are able to register touch input but they cannot display anything.

After that, participants were asked to perform a number of actions and to think out loud while trying to complete them. Only in moments where the participants were stuck and in no way could execute a task, extra explanation was given in order to help them to progress with the task.

In order to familiarize participants with the idea of swiping, they were asked to perform two simple actions: go to the next song and pause the song. This was asked so that the participants started with two easy and similar actions, swiping left and up. By default, the prototype displayed the media pane so participants only had to use the right touchpad to complete the task.

Next, they were asked to go to the navigation pane, navigate to home, and cancel the route. This involved using the left pad and custom gestures for the first time.

After that, they were given a broader instruction: reply to a message sent by Alice. This required the participants to find this information in the right pane (a WhatsApp message on the phone pane). In the end, they were asked to go back to the previous song. This action is very similar to the first one to see if they got used to the system and used a direct swipe, instead of using the menus to navigate.

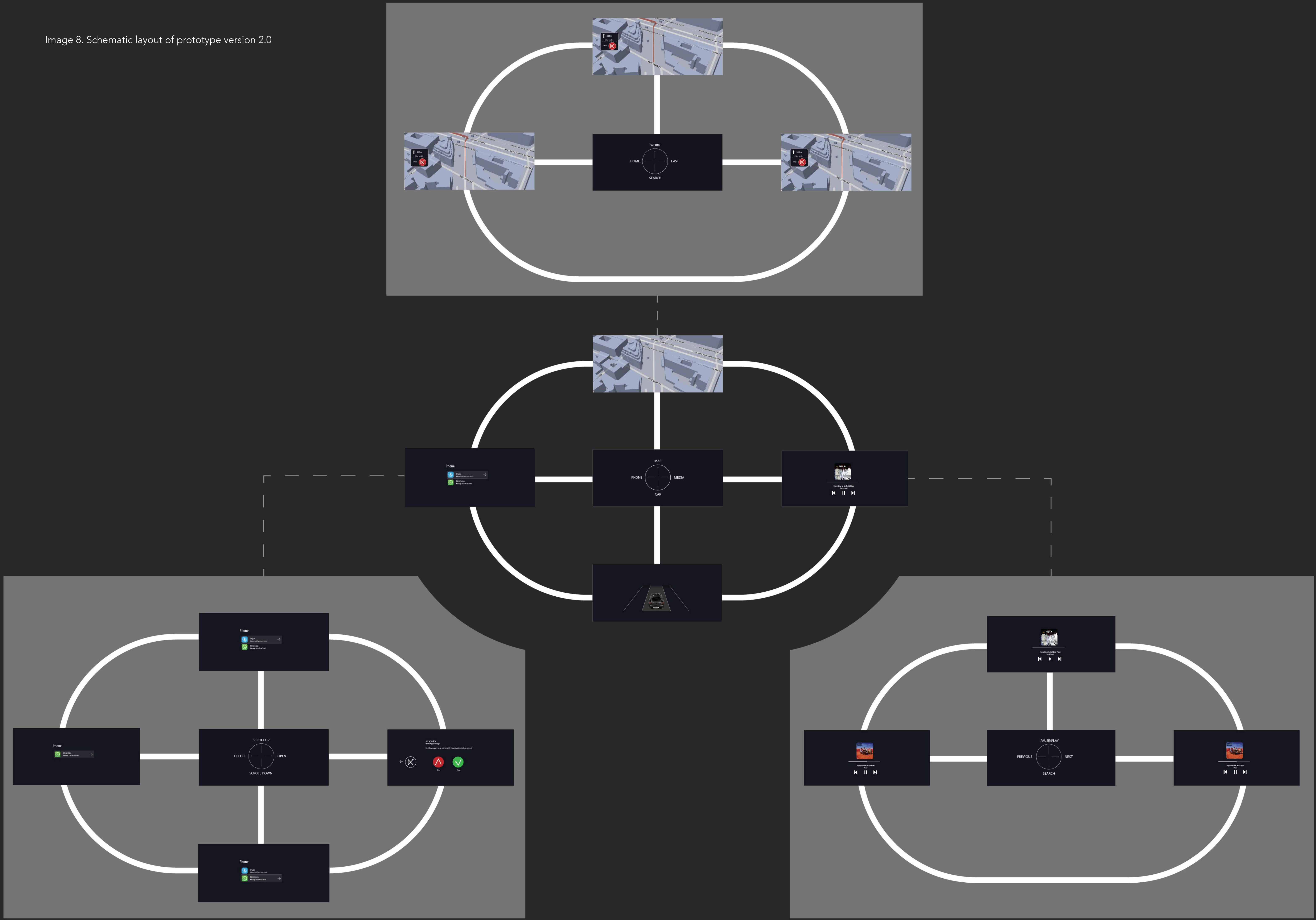
In case participants kept using the menus, they were asked to perform one more action: navigate to work. This is also an action very similar to the one they did before.

There were three main points to be found during the user test. First, do participants understand the menu structure with the main menu controlled by the left touchpad and the contextual menu controlled by the right touchpad? Second, can participants figure out that there is an expert mode (quick swipe without using the menu)? Third, can participants understand how and when to do the custom gestures on by themselves?

3.5 Results

In total 14 people participated in the user test, 6 of whom were UX designers. None of the par-

Image 8. Schematic layout of prototype version 2.0



ticipants had any previous knowledge about the project. As the system was running on an iPad, the test was performed in various locations inside Renault's design office.

To keep a record of the actions and thoughts of the participants, notes were taken during the tests (Appendix 1). After the tests, each problem and the number of occurrences were listed (table 1).

3.5.1 Test Results

The first task had mixed results. There were 5 participants, mostly UX designers, that intuitively swiped right on the right pad and completed the task perfectly. Even if this was the first time they touched the system. One of these participants mentioned the similarity to the smartphone: *"Music is exactly like it works on the phone, very intuitive"*. Others had to figure out which interaction to perform. That is when the interesting interactions happened. The results show that most participants first tapped one of the touchpads. As a result, the message 'touch and hold to see the menu' was shown. However, more than half of the participants didn't see, ignored, or misinterpreted the message. After that, the participants tapped the other touchpad or started swiping.

A remarkable finding on the second task, to pause the song, was that 10 of the 14 participants initially tried to pause the song by tapping on the right touchpad. Even if they had discovered the menu at that point. In fact, it happened in 6 tests that a participant discovered how to use the menu, but still instinctively performed a gesture that was wrong. For instance, when selecting a message on the phone, participants had to scroll down to the right notification and wipe right to open it. 4 participants used the left touchpad to scroll down. Presumably, because that is the touchpad they used most recent when navigating to the phone. 3 participants tried to open the message by tapping on the right touchpad.

On the other hand, there were 6 participants who used the expert mode to open the message without looking at the menu once to see what the controls are.

Next is the task that involved custom gestures for the first time: setting up and canceling a route on the navigation. During this task, most participants needed help to complete it. This can be attributed to several issues. First, the interface shows a representation of the gesture. 5 participants did not see this or misinterpreted it: *"I thought it was a Bluetooth icon, so I didn't pay attention to it"*. When the interface was understood correctly, 8 of the 14 participants tried to perform the gesture on the left touchpad instead of the right one: *"The icon is on the left so I thought I had to use the left side"*. In hindsight, this seems logical but it did not come up during the preliminary user test. In order to complete the task, the participants had to be told to look at the interface to see the gesture, or to

	Interaction Problems		Technical Problems
10	Tap to pause song	5	Tap is registered as swipe
8	Touch and hold message ignored/unclear	4	Quick swipe is registered as tap
8	Custom gesture on left pad	4	Two touchpads at the same time cause error
5	No menu feedback with custom gesture	3	Menu does not show up in navigation
5	Custom gesture not understood	2	No internet
4	Wrong pad used for scrolling	2	Users think they can press the whole iPad screen instead of just the touchpads
4	Expert mode not discovered	2	Nothing happens when using the search on the media page
3	Use search in media to go to map	2	Menu appears but doesn't disappear if no touch is registered
3	Swiping is difficult/feeld unnatural		
3	Tap to open message		
3	Selecting words instead of swiping		
3	Use left pad to cancel navigation		
2	Touch and hold is not clear (does other gesture to see the menu)		
2	Custom gesture looks difficult		
1	Menu structure is unclear		
1	Tries opposite gesture as undo action		
1	Disconnection between expert mode and menu		
1	Use circle gesture to scroll		

Table 1. Problems that participants encountered during the user tests listed by number of occurances.

use the right touchpad instead of the left one.

In 5 of the 14 user tests, the first thing the participants tried to do when being asked to cancel the navigation, was to hold down on the touchpad to see a menu. But no menu appears when doing the custom gestures.

The task where participants were told to reply to a message was expected to be more difficult than it turned out to be. After finishing the first tasks, almost all of the participants had a good understanding of what interaction to use at each point. 6 out of the 14 participants used the expert mode and some point during this task.

There was one problem that occurred 4 times during this task where participants used the left pad to scroll down. These participants understood the menu structure and which pad to use at which time before starting with this task. It suggests that they intuitively used the left pad: *"I used the left one for the last actions so I think that I don't think about the right one"*.

All 14 participants used the custom gesture without problems.

The last task was to go back to the media and back to the previous song. This was asked to see if the participants were used to the system and using the expert mode. 10 out of the 14 participants were using the expert mode to either go to the media pane or to go to the previous song. Though, one of the participants did it by accident. This person, together with the 4 participants who did not use the expert mode, and one other participant who seems like he did not yet fully understand the interaction, were asked to complete one additional task. They were asked to navigate to work, again, very similar to another task. Out of these 6 participants, 2 used the expert mode.

There was only one participant who needed help more than 2 times to complete the tasks. However, there were 4 participants, including the latter, who struggled throughout the test to interact with the system. 2 of these participants thought they had to do a specific gesture to see the menu. Even after explaining that a touch and hold is sufficient, they had problems adapting to that. 3 of the participants initially thought they had to hover over the words in the menu to select the function. Only after the user test finished, they realized that they had to swipe. On the other hand, there were only 2 participants who struggled to complete the custom gestures, arguably the most technical interaction.

3.5.2 Analysis

There were three important parts of the system that participants had to figure out: the menu structure, the expert mode, and the custom gestures.

Menu structure

As mentioned before, not all participants managed to find the touch and hold menus on their own. An important problem during the first task was software related. During 5 user tests, a tap was registered by the system as a swipe. This led to participants thinking that tapping was a way to interact with the system. This also happened when testing the first version of the system and as a result, a message was built in that shows that the system registered a swipe in a certain direction and not a tap. However, this was either not seen or ignored in those cases. These participants started out with an incorrect mental model of the system and some of them needed some explanation to discover the menu.

Also, the message that was displayed when participants tapped was often either ignored or misinterpreted.

These two points highlight the fact that notifications or message are not ideal for use in this context. More user testing has to be done to discover what the best way is to inform users about the interaction of the system when they first interacted with it. A possibility would be to use no text, but descriptive animations and icons. This is supported by the fact that when participants were demonstrated how to perform taps and swipes after the user test, all of them performed way better and more 'natural'.

Once the participants got the hang of the touch and hold, of the 14 participants, 13 figured out the general menu structure at some point during the test.

Expert mode

10 out of the 14 participants figured out the expert mode at some point during the test. A trend that can be seen in the test is that the participants who figured out the swipe interaction early on, including the expert mode, made very few mistakes with the rest of the navigation through the system. On the other hand, participants who started off with a different mental model of the system and struggled to complete the first tasks kept performing slower even though they completely understood how the system worked. Though, when asked to perform extra tasks at the end of the test or when playing with the system after the test ended, they did improve. They understood the swiping in 4 directions and the expert mode and were able to use without problems.

Surprisingly, only 4 people did not manage to either discover or understand the expert mode. These were the same 4 participants as described above, who seemed to struggle in general with the interaction.

Having 4 participants who struggle with the general interaction of the system is concerning. However, after the user tests, they were given more explanations and they were shown how to complete the tasks. After that, each of the 4 participants performed considerably better. That suggests that with better instructions, they would have completed the tasks more easily. Animations and icons could be the way to do this.

Custom gestures

The discoverability of the first custom gesture in the test also proved to be difficult for participants. Two main issues were observed. First, some participants were used to open the menu for each task that they did not know how to complete. However, the custom gestures had no menu or instruction when the participants touched and held down on the touchpad. This would be a great point of improvement. If an instruction was displayed to tell the user how to do a custom gesture, they would have probably been able to complete the task without help. The other point is that the interface was not clear enough. First of all, most participants did not notice the icon of the custom gesture. Second, the icon was displayed on the left side of the screen leading participants to think that the custom gesture should be performed on the left touchpad.

These two problems led to 13 of the 14 participants getting stuck in completing this task. A possible way to avoid this would be by improving these two points: 1. displaying an instruction to explain to users how to use the custom gesture and 2. moving the icon to the same side as the touchpad that participants need to use to perform the gesture. Once the participants passed the task with the first custom gestures, they had very few further problems with the other custom gestures.

Even though many of the participants needed help at some point in their test, most of the issues resulting in that can be solved by providing better explanations of the system in the interface. For instance, a short animation can be shown that the user can imitate to complete the first interaction. Also, instead of a message, icons could be added as a way to help users easily interpret the explanation. Additionally, the interface could be adjusted to show elements relative to their interaction on the touchpad. For instance, the 'next song' button on the interface could be displayed on the right side of the screen, 'pause' could be displayed at the top, and previous on the left. This way, users wouldn't always have to go use the menu to see which direction to swipe.

Additionally, a software improvement related to the gesture recognition is advised as a way to avoid issues encountered by participants during the first task.

04

Conclusion

This project tried to answer the question: *if it was possible to design the interactions between a person and a car from scratch today, how could they be designed?* Considering this, new interaction models were explored. Building on academic research, a broad concept, consisting of three parts, was designed to show a new way of interaction with the car using existing technologies.

Gesture interaction on the steering wheel was one of the three parts. Touchpads on the steering wheel are a new concept in the automotive industry. Therefore a user test was set up to explore one aspect of this concept: discoverability. The test highlighted a high discoverability potential of gesture interaction. When users were presented with a steering wheel with two touchpads, a simple interface, and basic instructions, they didn't need much time to figure out how the system worked. Most of the participants learned to use the system very quickly after trying to perform the first or second task. However, most of the participants did need help at some point of the test to complete certain tasks. Based on these observations, two main points of improvement are suggested to ease the process for users:
First, the interface should be more focused on providing clarity for gesture interaction. There should be a clear link between the touchpad and the interface that instructs the users what to do at each stage.
Second, the instructions on the screen should be more demonstrative in a way that users understand more easily the type of interaction needed from their side: when and how to do a swipe gesture, and when and how to do a custom gesture.

A lot more user testing is needed to determine if this type of interaction is valid. First of all, it would be interesting to test the suggested improvements. Second, a logical next step would be to make a prototype in a real cockpit of a car. The most crucial elements that should be tested are the usability and the effect this interaction has on distracted driving. For instance, what kind of interface should be shown to the users. Also, what kind of gestures should be used and whether, and to what extent, users are able to define their own gestures.

Testing the other concepts that were presented fell beyond the scope of this project. Making prototypes of these would be valuable, especially the natural interaction as it is largely based on academic research outside of the automotive context. The increase of ADAS technologies also highlights the need for this as users should be informed correctly about what the car senses and the decisions it makes.

On a broader level, this research has highlighted the need to move away from traditional interaction models towards different, more modern ones. In recent years, there's been a significant increase in new technologies being incorporated into the interior design of cars. However, the interaction models remain the same.
In a landscape where car startups are offering new user experiences in the car, where new business models are taking over, where major technology companies are moving into the car industry, and where cars are becoming more and more autonomous, the need for a simple and effective user interaction is more important than ever. This project shows the first steps in achieving that.

Acknowledgements

First of all, I would like to thank Marc Pinel for giving me the opportunity of doing my internship at Renault, and for giving me the freedom and help for completing the project.
Next, I would like to thank James Eagan for supervising me during the project. Also, I would like to thank the participants in my user test, who took time out of their busy days to give me their valuable insights and to help me further with my project.
Last, I would like to thank my girlfriend for proofreading my thesis and providing me with her valuable insights throughout the project.

Bibliography

[1] Leonardo Angelini, Francesco Carrino, Stefano Carrino, Maurizio Caon, Omar Abou Khaled, Jürgen Baumgartner, Andreas Sonderegger, Denis Lalanne, and Elena Mugellini. Gesturing on the steering wheel: a user-elicited taxonomy. In *Proceedings of the 6th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, pages 1–8. ACM, 2014.

[2] Olivier Bau and Wendy E Mackay. Octopocus: a dynamic guide for learning gesture-based command sets. In *Proceedings of the 21st annual ACM symposium on User interface software and technology*, pages 37–46. ACM, 2008.

[3] William Buxton. *Less is more (more or less)*. na, 2001.

[4] Tanja Döring, Dagmar Kern, Paul Marshall, Max Pfeiffer, Johannes Schöning, Volker Gruhn, and Albrecht Schmidt. Gestural interaction on the steering wheel: reducing the visual demand. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 483–492. ACM, 2011.

[5] Johan Engström, Emma Johansson, and Joakim Ostlund. Effects of visual and cognitive load in real and simulated motorway driving. *Transportation research part F: traffic psychology and behaviour*, 8(2):97–120, 2005.

[6] Focus2Move. Global car sales by manufacturer: The 2017 final count, 2017.

[7] Iván E González, Jacob O Wobbrock, Duen Horng Chau, Andrew Faulring, and Brad A Myers. Eyes on the road, hands on the wheel: thumb-based interaction techniques for input on steering wheels. In *Proceedings of Graphics Interface 2007*, pages 95–102. ACM, 2007.

[8] Joanne L Harbluk, Y Ian Noy, and Moshe Eizenman. The impact of cognitive distraction on driver visual behaviour and vehicle control. Technical report, 2002.

[9] Kristin Kolodge. Driver interactive vehicle experience (DrIVE) report, 2015.

[10] Dave Lamble, Tatu Kauranen, Matti Laakso, and Heikki Summala. Cognitive load and detection thresholds in car following situations: safety implications for using mobile (cellular) telephones while driving. *Accident Analysis & Prevention*, 31(6):617–623, 1999.

[11] Sharon Oviatt. Human-centered design meets cognitive load theory: designing interfaces that help people think. In *Proceedings of the 14th ACM international conference on Multimedia,* pages 871–880. ACM, 2006.

[12] Sharon Oviatt, Rachel Coulston, and Rebecca Lunsford. When do we interact multimodally?: cognitive load and multimodal communication patterns. In *Proceedings of the 6th international conference on Multimodal interfaces*, pages 129–136. ACM, 2004.

[13] Bastian Pfleging, Stefan Schneegass, and Albrecht Schmidt. Multimodal interaction in the car: combining speech and gestures on the steering wheel. In *Proceedings of the 4th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, pages 155–162. ACM, 2012.

[14] Groupe Renault. Global sales results report 2017, 2017.

[15] Raafat George Saad´e and Camille Alexandre Otrakji. First impressions last a lifetime: effect of interface type on disorientation and cognitive load. *Computers in human behavior*, 23(1):525–535, 2007.

[16] David L Strayer, Joel M Cooper, Rachel M Goethe, Madeleine M McCarty, Douglas Getty, and Francesco Biondi. Visual and cognitive demands of using in-vehicle infotainment systems. 2017.

[17] Cambridge Mobile Telematics. Distracted driving dangers, 2017.

[18] Derek Viita. Android auto and carplay remain popular among owners, 2018.

[19] Jacob O Wobbrock, Andrew D Wilson, and Yang Li. Gestures without libraries, toolkits or training: a $1 recognizer for user interface prototypes. In *Proceedings of the 20th annual ACM symposium on User interface software and technology,* pages 159-168. ACM, 2007.

Appendix

1. User Test Notes

Subject 1

Next song
I am right handed so I use right hand to swipe to the right

Pause song
Tries to tap to pause, reads message and finds the menu
Pauses

Navigation
Touch hold on right pad, cant find the menu option
Tries other touch pad
goes up

Navigate home
Uses left pad, doesn't find option
Uses right pad
Slides home via menu
-> Figures out main menu structure with submenu

Cancel navigation
Tries to open the menu via tap and hold -> doesn't work
Thinks it is a bug
Misses the gesture in the interface

He has to be told to look at the interface.
He thinks the gesture looks complex
He executes the gesture flawlessly

Message
Goes to phone

Figures out that you can quick swipe
Bug with touch vs swipe -> swipes too fast

Tries to tap to open the message, doesn't work
Does the gesture -> swipe right

Previous song
No problems

General comments
Maybe some app's have a timer to go always back to media
Menu not as overlay but in the corner
Turn by turn confuses the user. Separate turn by turn from gestures

Subject 2

Is swiping before executing my instructions.
Hasn't seen the menu

Next song
Next song without any problems

Pause
Taps instead of swipe up
Swipes up to pause
Misses the touch and hold message
Figures out menu/submenu system
Still can't find the menu interface
I try to do opposite gesture (swipe down vs swipe up) to undo. but that is different move
Still ignores the message

Navigation
He finds it but he does not know he did
I have to tell him about the menu
After that, setting up navigation is no problem
Cancel is no problem

Message
Uses wrong pad to scroll down

Previous song
No problem

General Comments
Without graphical representation you imagine different menu structure (rolling carousel)
Having just two songs makes it seem like you switch between them

Subject 3

Change song
Swipes right on left pad
I only have buttons on the left in the steering wheel in my car so I use left one here
Swipes right on right pad -> changes song

Pause
Taps the left pad
I think right is for all interaction (he tapped left pad and swiped right on left pad, both did nothing)
Taps, Sees and reads notification out loud but doesn't do anything with it
Swipes up on right pad -> song pauses

Navigation
Figures out the menu after touch and holding
Goes to map

Navigate home without problem

Cancel
Does not look like the easiest gesture to do
Does the gesture without problems

Message
No problems
Quick swipe

Previous song
No problem, by now he is expert user
But bug with touch vs swipe

General comments
I did not find the menu at first, but after it was

very intuitive

Subject 4

Next song
Swipes instantly without thinking

Pause
tap instead of swipe
Sees menu
Pauses song

Map
Uses left pad
Taps registers as swipe
Not used to menu
Swipes to wrong side

Cancel navigation
Doesn't know which side to do gesture
Gesture is displayed on the left, but operated by right pad

Message
Scroll on wrong pad
Uses circle gesture to scroll
Swipe down
Swipe in
Replies to message

Previous song
No problems

General comments:
Music is like it works on the smartphone, very intuitive
Position of gesture on interface is misleading.
Sometimes the gesture is displayed on the left, but you have to do it with the right
Begin of graphical gesture is unclear
Tap instead of swipe for pause

Subject 5 (not UX)

Next song
Intuitively uses right pad. Swipes right

Pause song
Tries to tap the right pad. Sees the message but doesnt understand it apparantly
Tries to tap the left pad. Sees the message, but again does nothing with it
Swipe up on right pad

Navigation
Tries to use the right pad.
Sees the message and discovers the menu.
Uses left pad. Finds the menu and goes to navigation

Home
Uses the menu.
I figured it out. Left is main menu and right is submenu

Cancel navigation
Tries to use the left pad at first but half way realizes that it seems wrong. Executes wrong command
Goes back to map and home navigation.
Uses right pad to do the gesture

Answer message
Very fast -> causes bug of touch instead of swipe

Previous song
No problems

General comments:
It took a while to see the menu, it was a bit hidden
I was used to the menu structure of swiping in directions, so i was a bit confused by the custom gesture. But after I discovered it, I liked it.

Subject 6 (not UX)

Realizes immediately that there is a menu, but thinks you can access it by swiping down for a bit, instead of touching and holding

Next song
Uses right pad and finds it because he knew the menu. But swiping does not seem to go very natural (very fast)

Pause song
The user tries to move the line in the menu on top of the word 'pause' instead of just swiping up.
The same for play. The system recognizes this as 'swipe right' so the user is confused.

It has to be explained that the user has to swipe and that he does not have to select the words.

Navigation
The subject figures out the menu structure.
He selects the right option but struggles with the swipes (too fast, wrong direction, etc.)

Home
No problems

Cancel navigation
The user tries to find a menu. Accidentally swipes to the left and media opens.
He goes back to navigation.
He does the same thing again.

The user has to be told that he has to look at the interface.
I thought it was a bluetooth icon, so I didn't pay attention to it

The user uses the wrong pad to do the gesture.
He has to be explained that he has to use the right pad.
He does the gesture, but with a lot of concentration (very slow)

Answer message
No trouble finding and opening the message
Replying requires some concentration

Previous song
No problems

Navigate to work
The participant was asked to do another action to see if he fully understood the system.
Executed without problems

General comments:
After you explained the swiping to me, it was easy to understand except for the cancel gesture

Subject 7

Next song
Taps the right touchpad. The system registers it as a swipe but the participant did not see the message that said 'swipe right'. He thinks tapping is going to the next song.

Pause song
Tries to double tap on the right pad. Again, does not see message.
Tries to tap on the left pad.
He tries to tap the Renault logo. It seems that he has not realized that he can swipe, and that he can only do this on the touchpads.
He gets stuck.
It is explained that he can tap once on the right pad. He reads the message and figures out the menu.

Navigation
He uses left pad. He figured out the menu structure.

Home
No problem.

Cancel navigation
He does not understand. He tries to find the menu but it doesn't show up.
He tries multiple swipes and taps. He tries to write stop on the right touch pad.
A tip is given to look at the interface.
He still seems lost. He tries to drag the white line on top of the graphical representation of the cancel gesture.
An explanation is given about the custom gesture.
He performs the gesture flawlessly.

Answer message
Navigates to phone without problems.
He does not know how to scroll down and opens the wrong notification.
He performs custom gesture to get out.
He opens the right notification and completes the task.

Previous song
No problems but he is still using the menus to navigate.

Navigate to work
Still using the menu.

General comments:
I did not understand the icon for the custom gesture at first. After you explained it to me, it made a lot of sense but maybe it should be placed closer to the controls.
He figured out the quick swipe but he didn't use it because he did not learn the system yet.

Subject 8

Figures out the menu before even starting the experiment.
Uses two touchpads at the same time which causes a bug.

Next song
He figured out the menu but still tries to tap to go to the next song.
He uses the menu and goes to the next song.

Pause song
Tries to tap again, then uses the menu.

Navigation
First tries to use the search function on the right pad. He then uses the left pad for the first time and figures out the menu structure.

Home
No problem

Cancel navigation
He figures out the custom gesture but uses the wrong pad.
The icon is on the left so I thought I had to use the left side.

Answer message
No problems. He uses quick swipe but it seems he does not realize there is the system with quick swipe and menu.

Previous song
No problems but still uses the menu structure

Navigate to work
Still not using the quick swipe

General comments:
It would be nice if there was a way to not see the menu every time. (he is then explained the quick swipe). I did not notice that it was possible to swipe quickly because it showed me a message.

It would be interesting to see if it is possible without menu structure. Now there is a lot of information that is separate and the user needs to remember the structure.

Subject 9 (not UX)

Next song
Participant taps the left touchpad. He reads the message and opens the menu. He tries to find the next song option so he goes to media.
He then tries the right pad and finds the next song.

Pause song
He taps the right pad first and then finds uses the menu.

Navigation
No problem

Home
No problem

Cancel navigation
Tries to open the menu but it doesn't show up. He finds the icon and performs it with the left touch-pad.
He is corrected and used the right touchpad.

Answer message
No problem

Previous song
No problem, now uses quick swipe.

Navigate to work
Not asked.

General comments:
The message in the beginning helped but at first I didn't know what it meant exactly. Touch and hold. Maybe an icon or animation is easier to understand.

Subject 10 (not UX)

Uses touchpads in the beginning. Touches them at the same time. Menu appears and doesnt disap-pear. Explained that 2 touchpads at the same time cannot be used.

Next song
Tries to tap the right pad. Then quick swipe.

Pause song
Tries to tap again. This time, reads the message and finds the menu. Pauses the song.

Navigation
First tries the right pad. Then the left pad. Figures out the menu structure.

Home
No problem.

Cancel navigation
Tries to find a menu. Gets stuck. Bug occurs so restart. Tries the left pad. Goes back to navigation.
Thinks this is the way to do it.
Is told that there is another way.
Finds the cancel gesture in the interface but does not know what to do. Is told to copy the gesture.
Does the gesture and completes the task.

Answer message
No problems. Uses quickswipe to scroll.

Previous song
Uses quick swipe to go to media. And to go to previous song.

Navigate to work
Not asked.

General comments:
The custom gesture was very unclear. It seemed like it belonged to the navigation and not to the gestures. The rest was very nice, I could remember the structure quite easily.

Subject 11 (not UX)

Next song
Tries to tap on the rightpad. Acknowledges the existance of the message but does not read it. Does not figure out to swipe. First swipes than touches which causes problems.
Now figures it out.

Pause song
Manages to pause the song by swiping and touch-ing. Still not the right technique.

Navigation
Tries to go to search. Swipe and tap instead of tap.
Finds the left menu. Seems like the participant wants to select the words, instead of the swiping behavior.

Home
Has trouble differentiating between left and right pad. Seems like the last used pad, is the one that has the preference.

Cancel navigation
Cant see the gesture. Goes to media and kind of completes the task like that. Is asked to try again.
The custom gesture is pointed out. Tries to do the custom gesture on the cluster screen. Then uses the left pad instead of the right pad. Finally figures it out.

Answer message
Uses left pad with right finger
Doesn't figure out the menu structure. Keeps using the left pad for every action.
Doesn't figure it out at all, is explained how the menu works.
Trouble with doing the custom gestures.

Previous song

Still does not understand the menu structure. Another explanation is given. Now it goes faster.

Navigate to work
Goes easer. Understands the menu. Still uses the wrong swiping technique.

General comments:
It is better than the buttons in my car because I don't understand these. I think I can get used to this.

Subject 11 (not UX)

Starts tapping before starting the test. Does some accidental swipes. Realize swiping is the way to control the system.

Next song
Tries to tap. Sees the message. Ignores it. Swipes to the right. Completes the task.

Pause song
Tries to tap again. Sees the message. Ignores it.
Tries to swipe with the left pad. Changes to car.
Tries to swipe in all directions until back at media.
Plays with this. Uses the right pad again but gets stuck.
Is told to read the message that says touch and hold.
Tries to hold down and finds the menu.
Pauses the song.

Navigation
Goes to navigation. No problem.

Home
No problem trying to find home.

Cancel navigation
Tries to find the menu. Doesn't show up. Tries left menu, shows up. Goes to navigation. Completes the task in the wrong way.
Is asked to do it again.
Finds the stop gesture on the screen but doesnt know what to do.
Is told to do a custom gesture.
Does the custom gesture on the wrong pad.
Next does it on the right pad.

Answer message
Goes without problems.

Previous song
Goes without problems. Is now quick swiping

Navigate to work
Not asked.

General comments:
The stop gesture was very unclear at first. After I did the message thing I understood it and it made sense.

Subject 12 (not UX)

Next song
Uses two touchpads at the same time. Causes error.
Has no idea what to do.
Is told to use the two touchpads.
Taps the right touchpad, is registered as a swipe.
Participant is confused.
Tries to tap again, this time the message is shown but ignored.
Asks for help. Touch and hold is explained.
He finds the menu. It seems like he wants to select the words.
Completes task.

Pause song
Uses menu again, tries to select the word instead of swiping up.
Completes task.

Navigation
Uses the search of media to go to navigation.
Nothing happens.
Uses left pad and finds the map. Still not swiping correctly.

Home
First uses the left pad. Then uses the right pad. He figured out the menu structure.

Cancel navigation
Tries to find a menu. Doesn't show up. Then uses left pad. He is confused.
He finds the gesture on the interface on his own.
Uses the wrong touchpad.
Is told to use the right touchpad.
Completes the task.

Answer message
Tries to scroll with left pad. No other issues. Swiping still seems difficult and he still seems to select words.

Previous song
No problems.

Navigate to work
No problems but no quick swipe.

General comments:
It is difficult at first. But after a while it became clear. With some more time I would figure it out.
It will be difficult in the car because you move the steering wheel and I had trouble to do the gestures with my thumbs.

Subject 13 (not UX)

Figures out touch and hold but not because of the message. Swipes a couple of times. Figures out he has to swipe. The system is reset for the start.

Next song
Swipes right on the right pad intuitively.

Pause song
Tries to tap the right pad. Message is ignored.
Tries to tap on the left pad. Message is read.
Finds the menu again and goes to media.
Uses right pad.

Navigation
Uses search instead of left pad.
Is confused because nothing happens.
Used left pad.
Completes task.

Home
No problem.

Cancel navigation
Tries to open the menu. Doesn't happen.
Does not see the customer gesture icon.
He uses left pad to stop navigation.
Custom gesture is explained.
He uses wrong pad, struggles with the gesture.
He fails to do the custom gesture on the right pad.
Tries again and completes task.

Answer message
No problem finding the message. Scroll with wrong pad. Tries to tap to open the message.
Custom gesture is slow but works

Previous song
No problems but still using menu instead of quick swipe

Navigate to work
Still uses menu. no problems

General comments:
The custom gesture is very difficult. I never did it before.
I like the menu structure.

Subject 14 (not UX)

Next song
Confused. Does not touch anything. Seems afraid to do something wrong.
Taps the right pad but ignores the message.
Taps again.
Taps the left pad.
Taps and holds a bit. Sees the menu.
Tries to find next song in wrong menu.
Uses right menu and finds song.

Pause song
Taps with two pads. Causes error. Restart the system.
Taps right pad. Then uses menu.

Navigation
First uses right menu. Then left menu.
Finds it without problems.

Home
No problems

Cancel navigation
Finds the gesture but uses wrong pad. And struggles with the gesture (wrong way around).
Does the right pad.

Answer message
Finds the message. Scrolls down on wrong pad. I used the left one for the last actions so I think that I don't think about the right one.
Taps to open the message. Then uses quick swipe.
Custom gesture now no problem

Previous song
Uses menu. Then does quick swipe a bit too quick.
Then he manages the quick swipe.

Navigate to work
Not asked, he figured out quick swipe.

General comments:
The custom gesture was difficult in the beginning.
Maybe for me, it is difficult while driving because you move your hands but it is very simple to use.