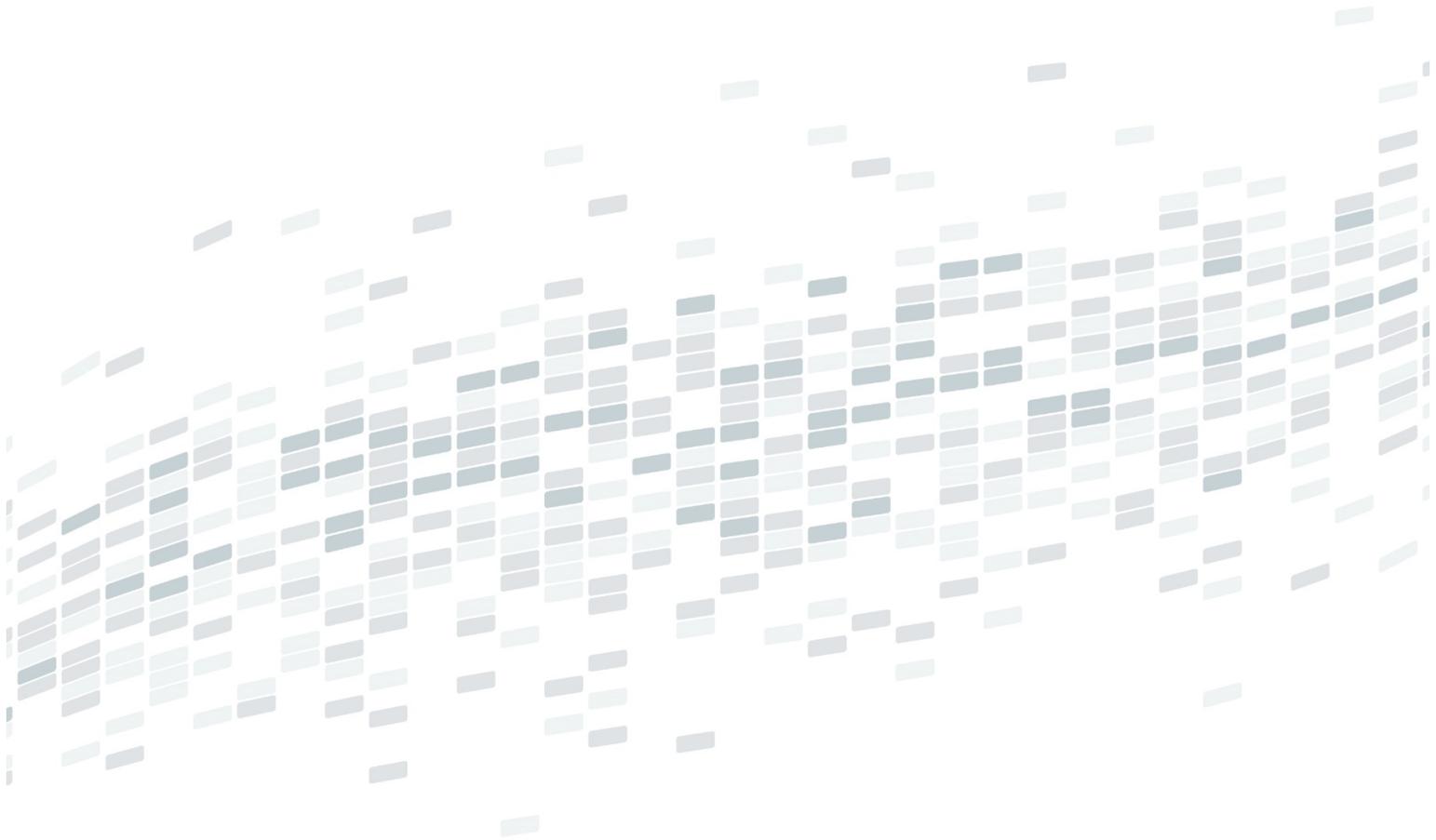


Visteon®

Complementing
Autonomous Driving with
Teleoperation:
Methodology,
Challenges and Best
Practices



Complementing Autonomous Driving with Teleoperation: methodology, challenges and best practices

Lucas Buckland, Basil Jesudason

Ankit Kaushik, Daniel Baudisch,
Andreas Wedel



by Visteon and Designated Driver

Introduction

From the rise in dockless vehicles to biometric sensors for healthcare to air quality sensing and geofencing that automatically compensates for poor air quality in certain locations, cities are growing smarter. Autonomous Vehicles (AV) are a critical piece of the smart city puzzle—but there are various use cases that require the safety net of a remote operator, such as when a water main breaks or a storm blows a tree across the road and a human operator must take over. Designated Driver (DD) offers real-time, human-operated control of AVs—essentially remote control of a vehicle minutes, miles or even oceans away. Remote teleoperation involves real world hardware running safety-critical software on real world streets with teams on different sides of the world. But what happens when a global pandemic hits?

This was exactly the situation that Visteon and Designated Driver faced when undertaking the “Highway Pilot” project, which aimed to integrate DD’s teleoperations software stack with Visteon’s open platform for autonomous driving, DriveCore™.

Project Overview

In the fall of 2019, DD and Visteon kicked off an ambitious project aimed at integrating DD’s teleoperations software stack with Visteon’s DriveCore. The project would culminate with an Audi A6 autonomous testing vehicle being remotely operated from Portland, Oregon, U.S., around a public road loop in Karlsruhe, Germany—close to the Visteon campus. It would utilize a combination of Visteon’s advanced driver assistance system (ADAS) software and DD’s teleoperations stack to navigate the complexities of the route. Visteon’s Highway Pilot, an ADAS feature, would be engaged when entering any highway sections. The teleoperator would observe the Highway Pilot, but not interfere unless necessary. When leaving the highway, the teleoperator would smoothly take control, and navigate the vehicle through the slower, but more complex, aspects of the route, such as intersections and roundabouts. What follows is a description of the Highway Pilot project, the challenges faced, the solutions proposed, and the things learned along the way; best practices in working remote and lessons and observations in integrating with an ADAS platform when it’s half a world away.

Remote Driving Complements Autonomous Driving: Use Cases

Despite the tremendous pace of autonomous development over the last few years, there still exist numerous situations where systems are unable to predict, or adapt to, complex scenarios. The integration of remote driving technology in an Autonomous Vehicle (AV) ensures a safety backup and alternative. We call it “the safety net for the autonomy system”. Remote driving becomes particularly useful in situations when an AV remains indecisive due to the complexity of the scenario, or when it encounters a hardware failure or sensor degradation due to bad weather conditions. Without this option, an AV would have to perform a minimum risk maneuver, which leads to a critical situation from a safety standpoint. Remote driving can act as a safe mechanism that allows us to bring the AV out from a critical situation, such as the one mentioned above. The collaboration between Visteon and DD was targeted to explore such scenarios, and how an autonomy system and remote driving system can work together to respond to them safely.

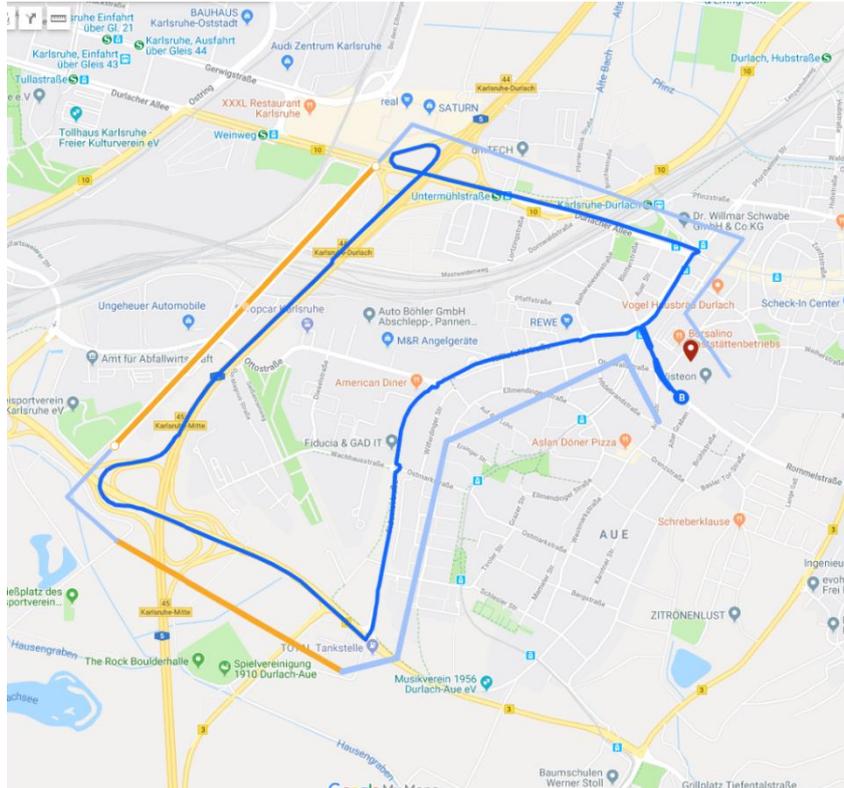
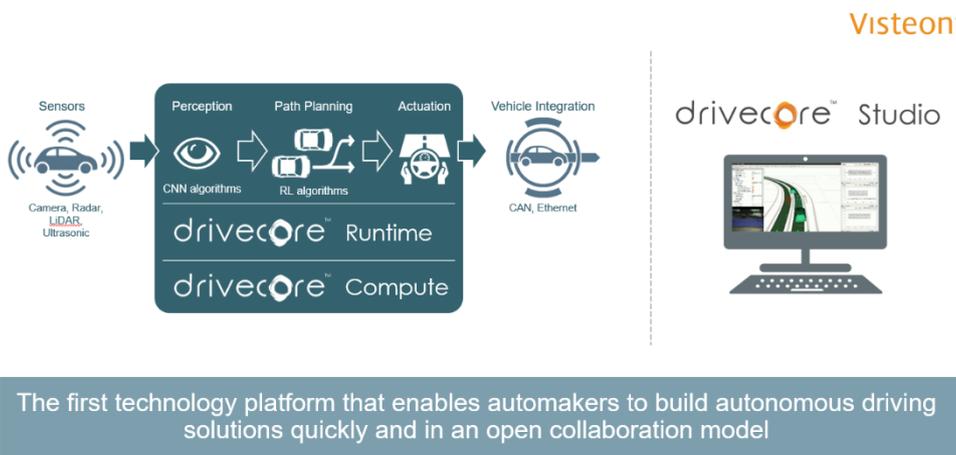


Figure 1: The test scenario in Karlsruhe, Germany showcasing a handshake between the teleoperation and autonomous driving features. The remote driver situated in Portland performs remote driving for the city driving section and the Highway Pilot feature from Visteon is operational on the highway and country road section. This is on an Audi A6 test vehicle located in Karlsruhe.

The Components



The first technology platform that enables automakers to build autonomous driving solutions quickly and in an open collaboration model

Figure 2: Visteon DriveCore

Visteon’s DriveCore is an autonomous driving platform which is open and scalable up to SAE Level 4 autonomous driving applications. The end-to-end solution incorporates three key components; DriveCore Compute, DriveCore Runtime and DriveCore Studio. DriveCore Compute is the modular and scalable computing hardware platform which forms the foundation of the platform. DriveCore Runtime

Working in Simulation



Figure 4: Virtual Test Drive Simulation

One of the big challenges that needed to be solved was how the teams would effectively test the software that was being developed. The testbed vehicle was in Karlsruhe and the control station in Portland. While the project would eventually culminate with DD engineers heading to Karlsruhe, it would be tough to do that prior to deployment due to COVID-19 travel restrictions. To ensure a smooth close to the project, things would need to be tested remotely. With DD's software it would be possible to drive the vehicle in Germany from Portland, but this requires a safety driver as well as a crew to get the vehicle to a testing location. Time zone differences were also an issue. How to allow for close to real-world testing, without being able to actually drive on the road?

The approach that was taken combined software and hardware in loop (SIL and HIL). The project would start with a simulator located in Karlsruhe that simulated the test vehicle, particularly the DriveCore's drive-by-wire interface. The simulator is equipped with a Highway Pilot feature from Visteon based on virtual test drive (VTD) VIRES simulation software. The first step in the project would be for Visteon to set up a VTD testbed in Karlsruhe, on the simulator or a SIL PC, that would facilitate the integration of DD's teleoperation. VTD would simulate the output of the four cameras that are placed on the test vehicle. An adapted DD Car stack would live on the same compute that was running VTD, in the Visteon office in Germany. It would allow a teleoperator to drive the simulated vehicle on a computer in Karlsruhe, from a control station back in Portland. To bring the simulation closer to real-world testing, a HD map of the test scenario presented in Figure 1 was integrated in VTD. In addition, DD Car hardware would be connected to the SIL in Karlsruhe, allowing the team to capture the round-trip-time delay for the overall system. This is measured from the camera input, to when it is video perceived at the remote driver station, to actuator inputs at the remote driver station and when the vehicle commands are looped back to the simulation.

The major interfaces to the system had been developed and tested, but there was still the issue of fully integrating DD Car with actual hardware and DriveCore Compute. Up until this point, the testbed ran on an x86-64 machine, with DriveCore Runtime installed. But DriveCore Compute is based on ARM, so the DD Car stack would need to be adapted to this and tested. Simulation was to be used to facilitate testing, leveraging some of the infrastructure already developed. A DriveCore Compute platform would be connected to the SIL PC used previously to allow communication between the DD Car stack integrated on DriveCore Compute and the drive-by-wire interface on the SIL PC. This would facilitate the DD Car stack running on the actual vehicle environment, with input from the simulator, replacing real world cameras and drive-by-wire feedback.

As a next step, the Highway Pilot feature would be enabled in SIL PC that would simulate ADAS features such as lane keeping, adaptive cruise control and automate lane change on highway and country roads. A remote assistance interface would be added in between the SIL PC and standalone DriveCore Compute that DD Car was running on, to allow the arbitration between Highway Pilot and DD Car's remote assistance functionality.

Finally, the system would be integrated with the test vehicle. The DD Car stack is then ready to go, since it has already been tested both in simulation and on public roads in Texas and Portland. Tuning is required for an actual vehicle, but this would be minor. Physical hardware would need to be configured to provide the drive-by-wire and video streams in place of the simulation. But DD Car is designed to be system and hardware agnostic so, other than minor tuning, this should be straightforward.

Teleoperation Results



Figure 5: Remote Driver station setup in Portland, OR

Figure 7 shows the remote driver station for driving the simulated vehicle that is running on the SIL in Karlsruhe. The video from the front camera is being streamed from Karlsruhe to the remote driver station in Portland where DD's software stack is used to control and drive the simulated vehicle.

Conclusion

Our teams learned the challenges of working remotely, how to address those challenges, and how to still develop a solid product. Some lessons; in communication, using the right tooling and transparency were reiterated. Other lessons were learned anew, like how to port to a new processor architecture without having that hardware available to test. It was all positive and helped us become better at what we do. We hope that other AV teams working remotely may find our learnings helpful.