PUTTING SAFETY FIRST IN AUTONOMOUS VEHICLES

StreetDrone's safety approach for testing and deploying connected and autonomous vehicles
Contents

A Safety First Approach to a Grand Challenge 4
Why Read StreetDrone’s Safety First Report 6
Introducing the Autonomous Vehicle Technology “Stack” 7
StreetDrone’s Safety Rules 8
Introducing StreetDrone’s Self-Driving System 9
Rule 1 - A “No Hack” Approach to Vehicle Control 10
Rule 2 - Single Point Failure Resilient 12
Rule 3 - Base Vehicle Dynamics Take Priority 13
Rule 4 - Clear Understanding of Control State 14
Rule 5 - Clear and Comprehensive Operational Design Domain Definition 16
Rule 6 - Protocols Based on Best Practice Safety Cases 18
Rule 7 - Safety Drivers Require Full Engineering Insight and Operational Training 19
Operational Safety Case Flow 20
Rule 8 - Errors Must be Traceable through all Aspects of the Autonomous System 22
Rule 9 - Collection and Transparency of Vehicle Data 24
Rule 10 - Testing Incorporates Simulation Prior to On-Road Development 26
Conclusions and Looking Forwards 28

The content provided here is a conceptual summary only, and does not constitute advice, nor is intended to be interpreted as a complete safety case. Material is subject to change and regular update.

This document contains links to external third party sites. These links are being provided as a convenience and for informational purposes only; they do not constitute an endorsement or an approval by the StreetDrone of any of the products, services or opinions of the organization or individual. StreetDrone bears no responsibility for the accuracy, legality or content of the external site or for that of subsequent links.

Copyright © 2020 StreetDrone Ltd. All rights reserved.
A Safety First Approach to a Grand Challenge

StreetDrone is on a mission to enable commercially-viable autonomous transportation services to be launched in urban centres around the world in the safest way possible.

We believe that the heart of cities represents the biggest opportunity for connected and automated vehicles. However, testing new autonomous mobility use cases in these urban areas presents significant safety challenges and considerations.

Our team’s combined automotive engineering experience has enabled a best practice approach to vehicle safety, prioritised at all levels of our technology deployment. This report details that approach in the hope that together, we can move forward in our discussions and understanding of what a truly safe autonomous future looks like.

About StreetDrone

Headquartered in Oxford, UK, StreetDrone is an autonomous technology business working on a full-stack solution for autonomy, from the conversion of vehicles to be ‘autonomous-ready’, through to the development of the self-driving software that helps those vehicles navigate their way around urban environments.

StreetDrone was the first business in Europe to run a public road autonomous trial using open-source self-driving software. We continue to develop technology that lowers the cost of autonomy in order to make it faster and safer for cities to deploy autonomous vehicles at scale.

Autonomous urban transport services will make our cities safer and healthier and have a significant effect on the lives of all those that travel around the world’s cities.
Why Read StreetDrone's Safety First Report

Safely deploying autonomous vehicles onto city streets is a complex, challenging and exciting area of technology development, and one which we think needs more focus. Safety must be permanently front of mind in all automotive product design and autonomous vehicles are no exception.

In the past few decades, vehicle technology has advanced from simple mechanics with little in the way of software integration, to now having sophisticated software to help drivers with all aspects of driving including managing engines as well as to brake and steer safely. Features such as lane-keeping, adaptive cruise control, self-parking and emergency braking mean that functionally safe software systems are now a mainstay of the automotive industry.

Adding the ability to allow a vehicle to drive itself means adding at least 10x in additional technology, all of which must obey strict safety criteria. The problem is that in this fast-moving world where being first to market is often rewarded with significant investment, many of the necessary and fundamental automotive safety considerations simply haven’t been followed.

In March 2018, an autonomous Uber test vehicle was traveling at 39 mph in Tempe, Arizona when it hit and killed Elaine Herzberg. Amongst a series of errors in the system design and deployment of their test vehicles, a subsequent investigation\(^1\) showed that Uber had switched off Volvo’s own safety systems, such as the emergency braking system, which would have made the accident much less likely to happen. It was found that one of the reasons for doing this was to ensure that Uber’s self-driving system and Volvo’s safety systems didn’t interfere with each other.

The accident shone a light on the lack of safety considerations of at least one of the biggest players in the burgeoning self-driving market. It was also avoidable.

This report is our take on what it means to be “safe” when it comes to deploying this new technology on our roads. It is by no means exhaustive, but it does address many of the issues that could have been implemented by Uber in Tempe in 2018. We have focused much of this report on the automotive standards that need to be followed in safely deploying converted vehicles for autonomy, an area largely ignored by many in the industry. This is something that will lead to more incidents if not addressed, potentially delaying the roll-out of a technology that should make our cities cleaner, safer and more enjoyable.

---

\(^1\) referring to the US National Transportation Safety Board’s Report (NHTSA-18-010) of 19th November 2019
StreetDrone’s Safety Rules

StreetDrone has distilled its approach to the safe deployment of autonomous vehicles into ten key rules which we believe define a best-practice approach to the roll-out of safe autonomous vehicles.

A key differentiator for StreetDrone is that the vehicle’s control systems are considered to be an integral part of the overall autonomous safety system, an area often overlooked by those solely focused on self-driving software. All of StreetDrone’s vehicles are adapted from road-going cars by adding additional components, like steering and braking motors, vehicle sensors, cameras, a computer, data networking and 4G/5G/WiFi communications systems. Our 10 safety rules, though not exhaustive, help us define how these vehicles are engineered and then deployed on the road.

Each of the 10 safety rules below is covered in detail as part of this report, with infographics to aid the reader in understanding our approach. Click on the rule number to jump to the additional detail.

Rule 1 - No hacking, spoofing or reverse engineering of the base vehicle control systems (those installed by the OEM) when adding our drive-by-wire system for autonomy

Rule 2 - No single points-of-failure in the autonomous drive-by-wire system, including in the sensors used to monitor the system

Rule 3 - Base vehicle safety systems, like emergency braking, always take precedence over self-driving commands

Rule 4 - The control “state” of the vehicle - whether the self-driving software or the safety driver is in control - must be clear and understood by the safety driver

Rule 5 - Comprehensive vehicle deployment protocols are based on best-practice published safety cases

Rule 6 - Clearly defined operational parameters, for instance where geographically, any vehicles are able to travel, are published and include all aspects of safety driver responsibilities

Rule 7 - Safety drivers require full engineering insight and understanding across the whole vehicle system

Rule 8 - Errors must be traceable through all levels of the autonomous system’s data, including at a vehicle level

Rule 9 - Collection and transparency of vehicle data must be compliant with the UK Department of Transport’s Automated Testing Code of Practice

Rule 10 - Autonomous software testing incorporates comprehensive simulation of real world scenarios

Introducing StreetDrone’s Self-Driving System

In order to deploy autonomous vehicles on the road, StreetDrone build our own drive-by-wire vehicle control system (called XCU) into standard road vehicles - currently into the Renault Twizy and Nissan e-NV200. This involves mechanically adding motors and motor controllers to the steering and brakes, as well as allowing the throttle to be controlled remotely, in order to replace the physical interaction between the vehicle and the driver when running autonomously. The system is made up this and the various other components shown here.

Vehicle control terms explained

Actuators - the motors used to control the steering and braking when in autonomous mode

Advanced Driver Assist Systems - features such as adaptive cruise control, automatic braking, automatic lightensing

Brake-by-wire - similar to controls added for learner driver cars, the system that provides a separately controlled motor which is added to the brake pedal system. The safety driver can affect a full mechanical override

CAN Bus - a standard communications network built into the vehicle, allowing electronic devices to communicate without bulky wiring looms

Drive-By-Wire - the system that electronically controls the vehicle’s functions, namely steering, braking and throttle, which would have typically been controlled by the driver

ECU - An electronic control unit (ECU) is any embedded system that controls electronic systems or subsystems such as a steering actuator

HMI - Human Machine Interface, the equipment fitted to allow the safety driver to interact with the vehicle, e.g. autonomous mode disable switch

SAE Level 2 - the SAE designation regarding safety driver interaction, meaning that they are fully engaged and monitoring the vehicle’s progress at all times. You can read more about SAE J3016 on the SAE website ODD - original vehicle manufacturer e.g. Nissan, Renault

Safety Driver - a driver tasked with monitoring the system who has knowledge of the vehicle architecture, software and has been fully trained in the operational safety and take-over procedures

Spoofing - Replicating original vehicle manufacturers control signals to manipulate vehicle function

Steer-by-wire - the system that, in addition to any powered steering present in the original vehicle, controls the steering via an additional motor and controller added to the vehicle when in autonomous mode. StreetDrone does not spoof the existing vehicle power steering motor, instead fitting an additional actuator

StreetDrone XCU - the ECU and wiring that run the StreetDrone Drive-By-Wire system

Throttle-by-wire - the control of the throttle request via electronic means directly from the StreetDrone XCU control system when in autonomous mode
Rule 1 - a "No Hack" Approach to Vehicle Control

At StreetDrone we have spent 3 years developing a sophisticated drive-by-wire (DBW) system that follows many of the best-practice safety principles detailed in this report. We recognised at the start of our development process that safety had to be the primary focus of the system, and using our significant experience in automotive control system design, including 20 seasons of insight derived from engineering design in sectors including Formula 1 and Formula E, Ian Murphy and Mark Preston began by defining the foundational principles on which we have based development ever since.

A drive-by-wire (DBW) system allows self-driving software to command the vehicle, to turn left or right, to go faster or slower, and to operate the horn, the indicators or lights. We build this system directly into a standard OEM vehicle and, in many ways the DBW system replaces the arms and legs of the driver, while the self-driving software replaces the brain.

The first of our rules is perhaps the most important in autonomy: that our drive-by-wire system works in unison with the original vehicle systems through the addition of motors on both the steering column and brake system, as opposed to using the vehicle’s existing control systems by “hacking” it. Additional motors mean that we remain independent of the control systems built for use in power steering or ADAS (Advanced Driver Assistance System) features, like automated emergency braking systems or lane-keep assistance. This ensures that we never take original OEM systems beyond the remit of their original testing or validation. In our safety analysis the alternative approach of leveraging existing vehicle systems carries a much higher risk, both technically and in terms of functional safety.

For instance, the hacking approach is limited in its applications, given that not every vehicle has the capability to function according to the drive-by-wire functional requirements, perhaps due to unsuitable actuators for this type of control system. This lack of suitability limits the range of vehicles that could ever be supported by hacking or reverse engineering.

Our decision to add a full drive-by-wire system comprises a large part of our safety concept, an analysis that influences all areas of our functional safety design. The analysis is based upon principles described in Functional Safety Standards such as IEC-61508 and ISO 26262, and consists of (among others) item definition (a description of the system, in particular its interfaces), environment definition (a description of the surroundings of the item), a hazard analysis for the item operated within the environment and the design of a safety concept that satisfies the safety goals.
Rule 2 - Single Point Failure Resilient

A key principle of the StreetDrone safety concept is that none of the vehicle safety mechanisms are dependent on the absolute reliability of any single component.

Our working assumption is that all hardware is potentially unreliable and could be subject to failure. If we make this assumption, we can avoid dependency on reliability statistics and ensure that our drive-by-wire system is hardware-agnostic.

The safety concept requires that the system is tolerant to all internal "single point failures". A single point of failure is part of a system which, if a failure occurs, will stop the system from working. This ethos requires all internal elements of the StreetDrone system to have a backup, ensuring that we can always reach a safe state in the case of any component failure.

StreetDrone has a two channel system - meaning one real time computer tells the system what to do whilst the other performs a monitoring role. Each has an independent set of sensors, and cross checks and verifies sensor data to provide error identification. In the case of an error, such as a sensor failure or a driver intervention via steering, throttle, braking, or cut-out switch, the system disables and passes control back to the driver. The pass-back to the safety driver is what we describe as our "safe state".

Unpacking StreetDrone's Safe State

The two-channel hardware implementation in each vehicle is coupled with an extensive array of sensors to monitor the status of each of the drive-by-wire actuators and their required position and behaviour. In each case, we employ multiple sensors to ensure operational redundancy in the drive-by-wire system.

The redundancy concept is a straightforward one; every sensor in the vehicle's control system is duplicated and remains in constant conversation with each other in what is termed 'controller' and 'supervisor' roles, meaning any discrepancy or difference in measured inputs and outputs, or internal message, automatically places the system into our safe state.

For StreetDrone, this safe state means a clear and total return of control of the vehicle to the safety driver. The hardware "no hack" implementation approach that we employ means that this is easily achievable by disabling the drive-by-wire system and removing power to the motors we install. We also allow the safety driver to actively disable the drive-by-wire system, and we'll cover this later in our safety report (see rule 4.)

Rule 3 - Base Vehicle Dynamics Take Priority

There is a further layer of safety that we consider critical when testing relatively immature autonomous systems. In addition to the checks made on sensor outputs and our safe-state handover, we also automatically check that any self-driving software isn't asking the vehicle to do anything that wouldn't be expected in normal driving situations, for instance in asking for full steering lock at high speeds.

These types of software errors are checked for in StreetDrone's drive-by-wire system, providing an additional layer of safety checks that might not be picked up by AI based self-driving systems.

In order to enable this, we add a rules-based function to sense-check requests from the autonomous software to the steering and braking motors. To ensure vehicle stability we set limits on how much steering angle, brake pressure or throttle can be requested by the system, the rate limits set according to vehicle stability. We measure this stability as part of our vehicle testing program.

As a practical example, the drive-by-wire system rejects inappropriate steering requests, such as a full steering angle rate request at 30mph, by limiting maximum steering angle to a safe level of 50%, because allowing such a command in full would result in the vehicle becoming unstable.

In addition, the system will put itself into a safe state if power to any electrical component in the system is lost or CAN communications fail. StreetDrone's control unit has a time-out whereby if it does not hear from the self-driving software computer within a certain time interval (due to any number of issues including physical severance of the wire), then it will inform and handover control back to the safety driver.

Established engineering principles have been followed to detect failure of CANbus control communications:

- CAN messages are subject to a timeout diagnostic
- CAN messages defined by the system are protected by a Cyclic Redundancy Check (CRC) to indicate when data is corrupted, and an alive-count mechanism to ensure that the connection from drive-by-wire to computer is still valid.

This system requires correct communication with the vehicle to enable and maintain autonomous software control.
Rule 4 - Clear Understanding of Control State

The importance of the safety driver became starkly clear when an Uber autonomous car being tested in Arizona, USA, struck and killed Elaine Herzberg as she crossed a road, while pushing a bike, on March 18, 2018. Instead of being aware of the car’s system state, the safety driver was distracted by a mobile phone at the time of the accident.  

StreetDrone builds an advanced HMI (Human-Machine Interface) into each vehicle to ensure that the safety driver is able to perform their duties with complete clarity and within some simple rules.

- Rule 1: the safety driver is always aware of who (or what) is in control at all times
- Rule 2: the safety driver is always able to take control should they need to, with a rapid transfer to safety driver control
- Rule 3: the system is not easily activated by accident

For rule 1 we use a system of lights and sounds to indicate which mode the system is in and whether the vehicle is being controlled by safety driver or self-driving software.

For rule 2, to ensure that the driver is always able to take control, there are multiple intuitive methods of takeover, including inputs to steering, throttle, brake (just like a normal driving experience of taking over from cruise-control for instance) or an array of switches and power isolators.

For rule 3, StreetDrone fits a multi-action activation system to the drive-by-wire, making it almost impossible for the safety driver to accidentally engage autonomous mode. In each case, the system ensures a smooth transfer to our safe state (described in the previous section.)

Ensuring that unpowered means safe

The safety driver is always able to take control of the StreetDrone vehicle, and we’ve built in some basic rules to ensure this can happen, even when our system is unpowered.

Firstly, the steering motor we use has a small enough residual torque that the safety driver’s control of the vehicle is unaffected when the system is unpowered, and it is the same for the braking system. Additionally, because of the way that the braking system is designed, with a uni-directional linkage, the safety driver can brake (and over-ride the system) even whilst the system is active.

The throttle controller, when unpowered, reverts to the original vehicle’s pedal sensor, andadditional protection to ensure that no alternate sources of control are accepted.

These 3 points ensure that, when the safety driver takes back control, unexpected system behaviour is not a possibility.

1 US National Transportation Safety Board’s Report (HWY16MH010) of 19th November 2019
**Rule 5 - Clear and Comprehensive Operational Design Domain Definition**

The Operational Design Domain (ODD) is something unique to autonomous vehicles and defines the conditions under which each vehicle can operate. In recent months, the way that each ODD is defined, and the rigour under which the technical maturity of autonomous vehicles is being validated under each ODD, has improved significantly.

We are learning quickly that by strictly defining the ODD, each test and trial of self-driving is much safer and more easily deployable, providing a focus for the scope of validated self-driving functionality in simulation, in off-highway testing and on public highways.

There are many attributes to a comprehensive ODD, including geographic location, test route accessibility, maximum vehicle speed allowed, the weather and local environment features to name but a few. Whilst StreetDrone’s approach to deployment is focused on complex but relatively slow-moving urban environments, ODD’s can still vary greatly within those areas.

Ensuring that the ODD is fully defined is an important factor when deploying autonomous systems safely. Making it as clear as possible when the conditions fall within the defined ODD enables the safety driver to swiftly react and take control when required. It is important that the ODD is comprehensive and fully defined as it forms the basis of both the safety case and the foundation for hazard analysis. It also heavily influences the verification work for both hardware and software, aimed at ensuring the automated driving system is capable of performing all functionality within the defined ODD.

The ODD within which StreetDrone’s self-driving software and StreetDrone’s vehicles operate will evolve over time, as increasing maturity enables a wider range of addressable scenarios and operating conditions. Rather than attempt to solve the ultimate, complex ODD (i.e. a vehicle that is capable of going anywhere, in all conditions), we’ve chosen to start with simpler functionality within specific conditions and increase complexity over time. This may involve lower speeds, geofencing and other operational constraints, but importantly allows confidence to be built and functionality to be validated in achievable, focused steps.

In addition to strict ODD definitions, StreetDrone also uses a Vehicle Design Domain concept (VDD), which allows a “lock-down” of the equipment and software to be used in that particular environment on a specific vehicle configuration. The VDD represents the configuration of the vehicle and hardware, from the number of sensors, down to the version of Ubuntu installed on the computer. This is extremely important for version control and traceability and allows a specific safety case to be built and run.

**DEFINING THE CONDITIONS FOR AUTONOMY**

StreetDrone have a comprehensive approach to defining the scope of autonomous vehicle trials - starting and finishing with safety.

---

**DEFINING THE ODD**

We work with our customers to define the trial environment and conditions; for instance, which road features are present such as traffic lights or roundabouts? What hazards are present such as pedestrians or cyclists? In addition, we add environmental parameters - do we operate in the rain and what are the speed route limits?

---

**THE VDD - SOFTWARE**

The software modules are then selected to suit the particular autonomous purpose of the trial - this can include dynamic path planning, object avoidance, pedestrian behavior prediction and more.

---

**THE VDD - HARDWARE**

Once we know the ODD features and software functionality you want to operate with, we can define the hardware requirements for the vehicle. This covers all considerations from computational power to the required sensor sets.

---

**FUNCTIONAL SPECIFICATION**

The ODD and VDD, once defined, form the functional specification for your StreetDrone autonomous vehicle - what it can do and where it can go.

---

**SAFETY DRIVER TRAINING**

When anything occurs (weather changes, dynamic objects, etc) outside your ODD and VDD conditions, the safety driver will take control of the vehicle.

---

**SAFETY CASE DEFINITION**

With all these steps complete, a thorough safety case can be written for an autonomous vehicle trial or demonstration!
Rule 6 - Protocols based on Best Practice Safety Cases

Each and every autonomous vehicle deployment is accompanied by a comprehensive safety document outlining how vehicles will operate, any risks that we've identified and how we're working with local people and organisations to ensure the safest possible roll-out. We do this in using an expert team from the automotive industry, while following the guidelines of PAS1881.

The safety case is built on established engineering and safety frameworks and revolves around using the existing functional safety analysis alongside a detailed operational risk assessment, producing a clear set of operational guidance documents. These documents ensure that every deployment follows a strict set of guidelines, minimising risk across all elements of the autonomous trial. Having already completed public road trials, we have a library of disengagement data and potential hazardous events and this data, alongside a detailed route assessment of the proposed trial environment, informs our hazard analysis and risk assessment.

Another key part of building a comprehensive safety case is a detailed analysis of relevant best practices, for example BSI PAS1881 standards and the Highway Code in the UK. This is then written into the safety driver training and operational guidance alongside the verification testing report for the automated driving system to ensure full compliance at all times. Other legislation and compliance analysis for deployment within the UK includes reviewing the relevant sections of the Road Traffic Act and the Road Vehicles (Construction and Use) Regulations and the code of practice for testing automated vehicles. 1

The diagram on page 20 shows the process of building a comprehensive safety case to support a deployment of autonomous vehicle technology.

Rule 7 - Safety Drivers Require Full Engineering Insight and Operational Training

Safety Drivers in autonomous vehicles shouldn’t be just test drivers or system supervisors, they need to have a complete understanding of the entire autonomous system, from an engineering perspective.

The fact is, that as driving assistance systems in road cars become more sophisticated, we are asking more and more of the general public to monitor complex engineering systems. In autonomous vehicles we need to buck this trend.

Firstly we want to stress the importance of the safety driver having an in-depth understanding of all aspects of the platform including system design, the hardware in play and autonomous driving software being employed. This knowledge enables the safety driver to understand the capabilities of the system as well as the expected and unexpected behaviours of each and every component. Most importantly the driver must be fully aware of the Operational Design Domain (ODD). Knowing this gives the driver the critical knowledge of what the system is capable of handling, and determines when they should take over in an unfamiliar scenario. Some illustrative examples include whether or not the system can handle adverse weather.

As an additional benefit, StreetDrone considers the role of safety driving to be an important part of early stage development of autonomous systems and can prove to be an invaluable way for engineers to experience real world examples of software development challenges. In this way, we enable important feedback loops between safety drivers and engineers, to log and record behaviours that help improve core technology developments.

Secondly, StreetDrone’s Safety Drivers undergo extensive training conducted in controlled environments, prior to any public road deployments. These include:

- A review of relevant literature, including the DfT Code of Practice, relevant StreetDrone vehicle safety manuals and familiarisation with ODD documentation
- System induction and walk through of all vehicle controls
- A closed road / track induction
- An on-road induction performed with a senior safety driver

This rigorous training programme insures all safety drivers are confident with the system prior to any public road trials.

Finally, all StreetDrone autonomous deployments and trials adhere to a timetable designed to set maximum working time for safety drivers, usually 4 hours in any single day with significant breaks, in order for them to maintain attention and not suffer from fatigue. We are careful that safety drivers feel able to speak up with any concerns they may have and that each driver has the right to stop a trial should they feel the need to.
Operational Safety Case
How does StreetDrone assist in helping to deploy self-driving tech on the road?

Operational Design Domain
Definition
What speeds do you want to run your vehicle? What kind of weather will you encounter? What is the road structure?

Stakeholder Engagement
We'll work together with local landowners, CCAV, emergency services and local authorities including the Council, and ensure the community is on board.

Operational Risk Assessment
What mitigation can be put in place for potential hazards? What level of risk is there?

Route Assessment
Where do you want to stage your trial? What hazards are there on your route? What is the accident history of the area?

Operational Guidance
We'll help define a process for the trial operation, including systems checklists, data handling and emergency response protocols. You're good to go!

The key to gaining confidence is incremental functionality increases
"From here our target is to expand the ODD operational design domain, in a gradual manner, but always with our focus on urban zones and low speeds."

"We're looking to develop in terms of the complexity with which the system can cope. It's a gradual process and one of the really good things about starting off so slowly is that it gives you a proper awareness of the potential hazards, so you can develop the system to cope."

- Ross James, StreetDrone Lead Safety Engineer

Defining the ODD is key to a defensible safety case
"With a locked down Operational Design Domain, we can build out all of the other key areas of the safety case. By starting with simple functionality, we can achieve a comprehensive safety case, build trust and confidence, before working out a reasoned plan to increase the level of functionality."

- Ross James, StreetDrone Lead Safety Engineer
Moving from rules to deep learning

Rules-based robotics is currently the only way to certify vehicle actions on public roads in the UK, due to the legislative framework in place, including DIT compliance. But deep learning, a different way of programming autonomous vehicles to localise, plan and react, is an increasingly common approach taken by a growing proportion of the self-driving industry. Deep learning is broadly used in complex computer vision and perception applications.

In autonomous driving applications, deep learning can be used in two different approaches: semantic abstraction and disruptive end-to-end learning. On the first approach the algorithms are focused on one part of the task, such as lane detection, object detection and blind-spot monitoring. These components are then merged into a main network that calculates the final driving commands. On the second approach the deep neural network is responsible for the end-to-end learning of the vehicle. The car is teaching itself how to drive, using a huge dataset of actual manual driving data, all of which are simply not possible using rules-based robotics.

This brings a challenge for the near future in the UK: how to legislate for deep learning and AI approaches in terms of functional safety. StreetDrone are committed to being part of this process and are working with both local and national government to achieve these guidelines, on the way to commercial Level 4 vehicles (as per the SAE levels of Autonomy - no safety driver required in certain Operational Design Domains). In addition, our approach to ensuring base vehicle control safety through a robust drive-by-wire approach combined with highly trained safety drivers, allows testing of a deep learning based system to be done at a minimal risk.

StreetDrone’s high level software architecture

Sensors
- Cameras
- Radar
- Lidar
- GPS/IMU

Perception
- Detection
  - Static Obstacle Detection
  - Dynamic Object Tracking
- Free Space Detection
- Map Localisation

Planning
- Route Planning
- Trajectory Planning

Control
- PID
- MPC
- DBW Actuators

Moving from rules to deep learning

Rules based robotics is currently the only way to certify vehicle actions on public roads in the UK, due to the legislative framework in place, including DIT compliance. But deep learning, a different way of programming autonomous vehicles to localise, plan and react, is an increasingly common approach taken by a growing proportion of the self-driving industry. Deep learning is broadly used in complex computer vision and perception applications.

In autonomous driving applications, deep learning can be used in two different approaches: semantic abstraction and disruptive end-to-end learning. On the first approach the algorithms are focused on one part of the task, such as lane detection, object detection and blind-spot monitoring. These components are then merged into a main network that calculates the final driving commands. On the second approach the deep neural network is responsible for the end-to-end learning of the vehicle. The car is teaching itself how to drive, using a huge dataset of actual manual driving data, all of which are simply not possible using rules-based robotics.

This brings a challenge for the near future in the UK: how to legislate for deep learning and AI approaches in terms of functional safety. StreetDrone are committed to being part of this process and are working with both local and national government to achieve these guidelines, on the way to commercial Level 4 vehicles (as per the SAE levels of Autonomy - no safety driver required in certain Operational Design Domains). In addition, our approach to ensuring base vehicle control safety through a robust drive-by-wire approach combined with highly trained safety drivers, allows testing of a deep learning based system to be done at a minimal risk.

StreetDrone’s high level software architecture

Sensors
- Cameras
- Radar
- Lidar
- GPS/IMU

Perception
- Detection
  - Static Obstacle Detection
  - Dynamic Object Tracking
- Free Space Detection
- Map Localisation

Planning
- Route Planning
- Trajectory Planning

Control
- PID
- MPC
- DBW Actuators

Moving from rules to deep learning

Rules based robotics is currently the only way to certify vehicle actions on public roads in the UK, due to the legislative framework in place, including DIT compliance. But deep learning, a different way of programming autonomous vehicles to localise, plan and react, is an increasingly common approach taken by a growing proportion of the self-driving industry. Deep learning is broadly used in complex computer vision and perception applications.

In autonomous driving applications, deep learning can be used in two different approaches: semantic abstraction and disruptive end-to-end learning. On the first approach the algorithms are focused on one part of the task, such as lane detection, object detection and blind-spot monitoring. These components are then merged into a main network that calculates the final driving commands. On the second approach the deep neural network is responsible for the end-to-end learning of the vehicle. The car is teaching itself how to drive, using a huge dataset of actual manual driving data, all of which are simply not possible using rules-based robotics.

This brings a challenge for the near future in the UK: how to legislate for deep learning and AI approaches in terms of functional safety. StreetDrone are committed to being part of this process and are working with both local and national government to achieve these guidelines, on the way to commercial Level 4 vehicles (as per the SAE levels of Autonomy - no safety driver required in certain Operational Design Domains). In addition, our approach to ensuring base vehicle control safety through a robust drive-by-wire approach combined with highly trained safety drivers, allows testing of a deep learning based system to be done at a minimal risk.

StreetDrone’s high level software architecture

Sensors
- Cameras
- Radar
- Lidar
- GPS/IMU

Perception
- Detection
  - Static Obstacle Detection
  - Dynamic Object Tracking
- Free Space Detection
- Map Localisation

Planning
- Route Planning
- Trajectory Planning

Control
- PID
- MPC
- DBW Actuators
Rule 9 - Data Collection Must Conform to UK DfT Rules for Autonomous Testing

There are two principles that have guided StreetDrone’s data system design to this point. The first is that all errors should be fully understood, meaning that the engineering team need a stream of data from the entire autonomous system - from vehicle sensor data, to camera and LiDAR data, to data from the self-driving stack.

The second is that we make this data completely open to our customers, which is not always the case with some vehicle and system providers.

StreetDrone takes a three staged approach to data collection. In line with our safety-first principles we record both a government recommended data set, and also any other data that could assist our engineers in better understanding the systems behaviour.

The first method includes capturing all the information as recommended in the UK Department of Transport’s Automated Testing Code of Practice. This data is outputted through our data network via CAN and stored on a secure CAN data logging device within the vehicle. This CAN logger is configured such that it automatically starts to log data when the drive-by-wire (XCU) is powered on.

Secondly, StreetDrone records a more comprehensive set of data from all the sensors and software relating to the autonomous driving system, all of which is is collated into a single (rosbag) file. A rosbag is a common file format for storing message data on any self-driving system built on the Robot Operating System (ROS). This data can then be replayed to recreate the autonomous journey in real time making it easy to interpret and learn from. StreetDrone vehicles have the ability to store this data simultaneously on two separate hard drives within the vehicle, minimising the possibility of data loss.

Finally, StreetDrone uses dashcams in every autonomous vehicle. We use these primarily to capture video data relating to safety driver behaviour and actions, but also to record events that might impact the autonomous software behaviour, such as the behaviour of vehicles in front of the vehicle. Dashcams are particularly useful as they can provide instant and easy access to any footage of the safety driver and environment should any incident occur.
Rule 10 - Testing Incorporates Simulation prior to On-Road Development

Testing of autonomous vehicle software in simulation, before any tyres hit the tarmac, is an essential part of StreetDrone’s approach to software development. This applies for both StreetDrone’s proprietary software and any software customers wish to develop themselves.

The main purposes of simulation testing are to reduce physical testing time in the real world (this is often the most expensive part of developing autonomous vehicles), and to speed up development. It also provides the framework for validation of autonomous vehicle software and behaviour, and ideas around future legislation to prove the safety levels of the software, such as simulated ‘driver’ tests.

For StreetDrone, simulation work sits over the foundations laid by the functional safety within the vehicle control. It enables deployment on private proving grounds, followed by public roads, in an easy way for customers, by providing an extra layer of stability within the software development workflow.

Software Development Workflow

This workflow above for developing new features and improved functionality allows for full error-tracing, thorough scenario testing and detailed understanding of the problems to be solved. All these contribute to the functional safety of the system, and provide customers with confidence in StreetDrone’s proprietary software as well as their own developments.

Introducing CARLA

The backbone of StreetDrone’s simulation is based in CARLA, an open-source tool powered by Unreal engine. The benefits of CARLA over other robotics simulations include scalability (via the ability to run simulations on a remote server), a flexible API allowing customers to adjust traffic generation, varied and unexpected pedestrian behaviours, weather changes and more. CARLA is fully integratable with both ROS1 and Autoware, and forms the controlled environment in which software can be tested before running in the real world.

This workflow opposite allows for full error-tracing, thorough scenario testing and a detailed understanding of the problems that need to be solved. All these contribute to the functional safety of the system, and provide customers with confidence in StreetDrone’s proprietary software as well as their own developments.

We are currently developing a fully correlated vehicle model in CARLA, using data from real-world tests on StreetDrone vehicles such as four-post-hg tests. This comparison of physical data with simulation output is essential to prove that results in simulation will match outputs on a real vehicle in the real world; this all starts with a validated vehicle physics model, including tyres and suspension details.

This is followed by detailed scenario generation to develop localisation, perception and control modules, all in CARLA, using sensor models matching sensors on the real vehicle, including LiDAR, and cameras. In the future StreetDrone’s customers will be able to run their own simulations constantly in the cloud, generating thousands of data points and outcomes to examine different areas of the ODD and VDD. The SMLL (Smart Mobility Living Lab, a StreetDrone customer) test site environment has been re-created in CARLA, allowing for rapid development off-vehicle, saving valuable real-world testing time.

Regression testing is another part of the StreetDrone software development architecture. At the ECU level, StreetDrone’s regression testing tools provide the repeatability of development of the embedded controls systems, while Jenkins (a software delivery management tool) and other tools provide continuous integration for higher-level software development.
Conclusions and Looking Forwards

The autonomous vehicle market is still in its infancy, there's no doubt about that. The long-term viability of autonomous transport services remains unclear and the major automotive and tech players are still scrabbling around for an understanding of how this complex market will develop.

What constitutes a completely safe autonomous system is still unclear, but clearly defining operational governance should take us a long way. StreetDrone's approach is focused on creating a set of rules that can be built upon as a clearer understanding of use cases and service provision becomes available.

The good news is that there are an increasing number of organisations looking at safety in the industry. The recent release of PAS 1881 by the British Standards Institution involved a group of partners from the automotive and CAV industries including the Centre for Connected and Autonomous Vehicles, the Department for Transport, Innovate UK, TRL and Zenizc. PAS 1881 gives the industry a means to standardise how deployments of autonomous vehicles need to be documented and, although further development is needed to include more around vehicle technologies, this is a great start.

Ever improving standards of safety that support the deployment of connected and automated vehicles are essential for this industry to thrive. We look forward to working with players across the industry to come up with an ever improving set of rules to govern the deployment of autonomy into the world.

Thank you for reading this report and we look forward to hearing from those interested in continuing the discussion and debate around safety in the new autonomous world.

Mike Potts, Founder / CEO, and Mark Preston, Founder / Advisor

Contact

For further information, please get in touch via:
info@streetdrone.com
www.streetdrone.com

Copyright StreetDrone Ltd 2020. All rights reserved