Augmenting Self-Driving with Remote Control: Challenges and Directions

Lei Kang
University of Wisconsin-Madison
lkang@cs.wisc.edu

Wei Zhao
University of Wisconsin-Madison
wzhao@cs.wisc.edu

Bozhao Qi
University of Wisconsin-Madison
bozhao@cs.wisc.edu

Suman Banerjee
University of Wisconsin-Madison
suman@cs.wisc.edu

ABSTRACT

Self-driving or autonomous vehicle systems are being designed over the world with increasing success in recent years. In spite of many advances so far, it is unlikely that such systems are going to ever achieve perfect accuracy under all conditions. In particular, occasional failures are anticipated when such vehicles encounter situations not observed before, or conflicting information is available to the system from the environment. Under such infrequent failure scenarios, the research community has so far, considered two alternatives — to return control to the driver in the vehicle, which has its own challenges and limitations, or to attempt to safely "park" the vehicle out of harm's way. In this paper, we argue that a viable third alternative exists — on failure of the self-driving function in the vehicle, the system could return control to a *remote* human driver located in response centers distributed across the world. This remote human driver will augment the self-driving system in vehicles, only when failures occur, which may be due to bad weather, malfunction, contradiction in sensory inputs, and other such conditions. Of course, a remote driving extension is fraught with many challenges, including the need for some Quality of Service guarantees, both in latency and throughput, in connectivity between the vehicles on the road and the response center, so that the remote drivers can react efficiently to the road conditions. To understand some of the challenges, we have set up real-time streaming testbed and evaluate frame latency with different parameter settings under today's LTE and Wi-Fi networks. While additional optimization techniques can be applied to further reduce streaming latency, we recognize that significant new design of the communication infrastructure is both necessary and possible.

HotMobile '18, February 12–13, 2018, Tempe , AZ, USA

ACM ISBN 978-1-4503-5630-5/18/02... $15.00
https://doi.org/10.1145/3177102.3177104

KEYWORDS

Self-Driving Car, Remote Control, Live Streaming

ACM Reference Format:
Lei Kang, Wei Zhao, Bozhao Qi, and Suman Banerjee. 2018. Augmenting Self-Driving with Remote Control: Challenges and Directions. In *HotMobile '18: 19th International Workshop on Mobile Computing Systems & Applications, February 12–13, 2018, Tempe , AZ, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3177102.3177104

1 INTRODUCTION

A self-driving vehicle is one that is capable of sensing its environment and navigating itself without human input [23]. It uses a variety of techniques to sense its surroundings, such as LIDAR, RADAR, odometry, and computer vision. It uses these different sensor inputs to understand its environment, recognize various road conditions, traffic lights, road signs, lane boundaries, and track surrounding vehicles. The potential benefits of self-driving vehicles include increased safety, increased mobility and lower costs. It is estimated that self-driving vehicles can reduce 90% of the accidents and prevent up to $190 billion in damages and health-costs annually [11].

Many commercial and academic endeavors are putting significant resources for the development and tests of such self-driving systems [3, 14, 22]. For example, Google started its self-driving project in 2009, and has spent more than $1 billion in building and testing fully self-driving vehicles [12]. While legal and political challenges remain in its widespread adoption, there are also some technical bottlenecks on the way of developing completely reliable self-driving systems.

All self-driving systems make decisions based on the perception of the environment and predefined traffic rules. However, there has been occasional failures of these systems when they have encountered scenarios that were hitherto unseen. For instance, based on the situation in a construction zone, human drivers would realize that it is permissible to cross over a double yellow line by following the appropriately placed cones (which otherwise is illegal to cross in the US), while a self-driving vehicle may not be able to do so, and therefore be unable to move forward. Similarly in poor weather conditions or due to traffic light malfunctions, the cues from different sensors may contradict each other leading to confusion in decision making.

In general, the road rules are complex and may conflict with each other, i.e., the system has to understand when to follow cones and ignore lane markers, and when to obey a road worker and disobey traffic signs. We observe that the real world situations are so diverse and unpredictable, that there are always situations that cannot be matched with predefined rules and may occasionally cause self-driving system failures.

Hence, we propose to use *specially designated remote human operators to augment self-driving system when it fails to perceive or handle current situations.* It is well established that human drivers, especially experts, are capable of making good judgement calls in face of contradictory or inadequate inputs, that sometimes limit a learning system that has yet to encounter a scenario before. While it is tempting to return control (during the failure of the self-driving function) to a local human driver situated in the vehicle, it is foreseeable a future of driverless cabs carrying only underage or licenseless passengers. Hence, we propose to engage remote human drivers as a safety backup when the self-driving function fails. We expect that remote drivers can multiplex and manage a large group of vehicles making scalability feasible. To make such a remote response center with human drivers practical, many challenges and new research questions arise. For example, how the networking infrastructure and protocol should be designed to accommodate such safety and latency critical applications? Also, how the sensory data is processed and sent to the remote center? It brings up many human computer interaction and security issues as well.

As a start, we conduct feasibility study on a real-time streaming testbed and evaluate the impact of today's network latencies on streaming the vehicular environment through audio-visual methods. We illustrate that In today's LTE networks, it is usually possible to accomplish a two-way communication latency of around 100 milliseconds when streaming frames of various resolutions by using standard video compression algorithms [13], which is within the range of tolerant latency for racing games in previous studies [15]. To keep the latency of communication between the remote human drivers and the vehicles bounded, we anticipate that multiple remote response centers be created — based on the maximum end-to-end latency that can be accomplished in such future network designs. Each remote response center then is likely to only manage, control, and provide remote driving augmentation to vehicles within some vicinity. It is also possible that this function will be practical in denser areas with many vehicles, where such system failures are more likely to occur. In the rest of the paper, we explore various issues in realizing these goals and discuss both opportunities and challenges in this direction.

2 REMOTE CONTROL AS BACKUP

In this section, we illustrate how self-driving system works and under what conditions it may fail, e.g., the computer system cannot understand the semantic meanings of road/traffic

Figure 1: A fully self-driving system may fail to recognize the semantic meaning of "local traffic" and crossing double yellow line is allowed.

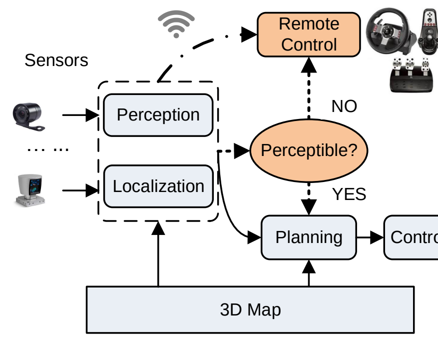


Figure 2: High-level architecture of self-driving system with remote control.

conditions. We posit that remote control system and human operator can augment self-driving systems in such conditions.

2.1 How Self-Driving System Works

A self-driving system consists of several modules that are responsible for perception, localization, planning and control [16]. A high-level architecture of self-driving system is illustrated in Fig. 2. Perception refers to the ability to collect information and extract relevant knowledge from the environment, such as where obstacles are located, detection of road signs/marking, and categorizing data by their semantic meanings. Localization refers to the ability to determine the vehicle's position with respect to the environment. Planning refers to the process of making purposeful decisions in order to achieve higher order goals, typically to bring the vehicle from one location to another while avoiding obstacles and optimizing over designed heuristics. Finally, the control module is used to execute the planned actions. Similar to human drivers, self-driving vehicle follows predefined traffic rules, such as drive within lane, do not cross double yellow line (except left turns), stop at the red traffic light (except right turns in certain cases) etc. The perception module identifies lane boundaries and traffic lights, and the the planning module makes decision based on predefined rules.

Table 1: Possible Self-Driving System Failures

Possible Failures	Examples
Perception failure at night and/or under challenging weather conditions	Unreliable camera at night (even with headlight); [1] Low visibility due to fog; [2] Lane markers are covered by snow.
Confusing or malfunctioning traffic lights/signs	Flashing yellow left turn light with sign instruction; [3] Malfunctioning traffic light turns to both red and green. [4]
Confusing detour due to misplaced cones or complex instructions	Instruction requires extra knowledge (e.g., local traffic only); the detour arranged by traffic barrels or cones is not clear (e.g., misplaced by road workers).
Complex and/or confusing parking Signs	Unclear, confusing, and handwriting road signs; [5] Parking is allowed only under certain dates and permits; Parking lots for particular vehicles, i.e., electronic or small vehicles;
Collision or system/hardware failures	Self-driving vehicle gets involved with collisions [22]; The LIDAR or other sensors fail.

2.2 Where Self-Driving System May Fail

A self-driving system may fail under complex road/traffic conditions caused by road constructions, traffic light malfunctions, randomly placed traffic cones, customized traffic signs, and many other conditions that can hardly imagine (examples in Table 1). One of such examples is illustrated in Fig. 1, the road is closed and only local traffic is allowed. The self-driving system could fail to understand the semantics of construction signs, i.e., the self-driving system cannot understand what "local traffic" means. Also, the system may fail to detour around the road construction zone, i.e., it has to understand when the self-driving vehicle is authorized to cross double yellow line or road boundaries. Also, the system may fail to detour around the road construction zone, i.e., it has to understand when the self-driving vehicle is authorized to cross double yellow line or road boundaries. The detour can also be arranged with traffic barrels which are placed by road workers. Since there is no specific rules to place traffic barrels, it is hard to find a general logic to learn where the detour is. It also has to identify road workers who is instructing the vehicles going through the road work zone. To verify a system with such capability, one has to navigate the system to drive through various road construction zones. In this case, a human operator can understand that the road is closed for "through traffic", and use external tools and knowledges to decide if the original route belongs to "local traffic". While it is possible to use map update and algorithm to handle this particular case, there are also other conditions that are so complex that a computer system may fail to handle and it takes years to realize various conditions and implement corresponding solutions. Some of the possible failures are summarized in Table 1.

[1] https://www.youtube.com/watch?v=uYav3_7miIc
[2] https://www.youtube.com/watch?v=fc0yYJ8-Dyo
[3] https://www.youtube.com/watch?v=56UDZLlj2q8
[4] https://www.youtube.com/watch?v=femUe6bds0U
[5] https://www.youtube.com/watch?v=cZfj9yL3cWk&t=108s

2.3 Why Remote Control

Remote control systems can act as an economic and safe backup of self-driving systems. One human operator can manage multiple self-driving vehicles and take actions upon request. It could also replace the human drivers prepared to take over the control after system failures [22]. A high level architecture is illustrated in Fig. 2. Suppose a road lane is closed due to road work, a self-driving system can simply detect that the current lane is blocked, while it may fail to understand the semantic meansing of the road signs. The remote human operator can take over the control if the self-driving system fails. Every detour is different and a fully self-driving system has to be trained and tested over millions (or even more) of such cases before it can be claimed as capable of fully self-driving. Remote control augmented self-driving systems could be more reliably handle such situations. Remote control will consume data for live streaming, but we argue that remote control is used only when self-driving system fails. In the example illustrated in Fig. 1, the live streaming and remote control is used only when the vehicle is taking the detour. It rises open questions such as how many camera feeds and what are the video quality requirements in this application scenario.

3 CHALLENGES AND ISSUES

The remote control system can be a safe backup of self-driving systems, but there are also challenges that should be addressed along the way. We discuss possible research issues and directions in this section.

3.1 Network Infrastructure and Protocol Design

To bound the latency between the remote human drivers and the vehicles, we anticipate that network infrastructures and communication protocol will be designed. In such a network, multiple remote response centers are created and each provides remote driving augmentation to vehicles within some vicinity based on network latency and bandwidth. The remote response center should be selected and/or switched based on the maximum end-to-end latency. In denser areas with many vehicles or during road constructions, this function will be more practical as system failures are more likely to occur. Given there are different wireless communication protocols, it rises open questions about how to ensure QoS under different network protocols. For example, how to optimize the transmission and ensure QoS under current LTE networks [6]? how to design 5G networks to ensure low latency and high bandwidth vehicle-related traffics [25]?

3.2 Perception Module Design

Similar to human drivers, self-driving vehicles make decisions by matching the contextual information with corresponding traffic rules. The self-driving system fails when the road/traffic condition is unrecognizable or multiple conflicting rules are derived. With remote control, the self-driving system can classify the road conditions into binary conditions: perceptible or not. To this end, the self-driving system can cache the 3D map of the road, traffic signs/lights and surrounding buildings [22], it can detect if it is perceptible by comparing perceived 3D map with cached version. By filtering out the moving objects, the two versions should be fairly similar. If the perception module detects road blockage or road signs that was not cached, then it tries to understand the situation, and transfers the control to remote human operator if it fails.

3.3 Real-Time Streaming

To provide sufficient information for the remote human operator to control the vehicle, the video streams and external contexts, such as extra camera feeds from the vehicle itself and other surrounding vehicles, should be sent to the server in real time. The question is what kind of data and in what format should be transmitted to the server? Suppose both the self-driving system and the server cache the 3D map of the surrounding infrastructures, only the differences can be transmitted to the server and the server can reconstruct the real-time conditions based on the differences. Also, optimization techniques can be used to compress and reduce the data volumes. The data size would be smaller if only the boundary of the objects are transmitted.

3.4 HCI for Remote Control

Remote control is different from control within the vehicle as it suffers various levels of latencies. The remote human operator may see delayed views and the vehicle also receives delayed control messages. There are many open questions like what kind of control is required from the human operator and what is the safe speed under current latency and remote operator's reaction time. The self-driving system can calculate several possible detours or alternative plans, and the remote human operator can pick the most appropriate one. In the cases where the self-driving system cannot understand the environment, a human operator can take over the full control of the vehicle. Self-driving system can also assist remote control. For example, while the vehicle is controlled by the remote human operator, the collision avoidance module of the self-driving system still works to ensure the vehicle will not get involved with any accidents. The reaction time of the remote operator, the latency of the networks and the vehicular speed should also be well studied to ensure the vehicle is at a safe speed.

3.5 Online Sharing and Learning

If one self-driving system fails, the scenario should be shared and used to train the system so that the self-driving system can better model or handle similar cases. It is also possible to store the road information in the cloud and the self-driving systems update with unusual conditions, e.g., lane closed for road construction, so that the vehicles passing by this road segments can be updated. Also, after the human operator makes a decision, e.g., detour around the blocked lane, other self-driving vehicles passing the same road segment should be able to follow the same detour automatically through online sharing and learning. In such cases, only one or few self-driving vehicles need to be controlled by remote human operators, and the rest self-driving vehicles can use updated road information and recorded decisions to pass through without further human inputs.

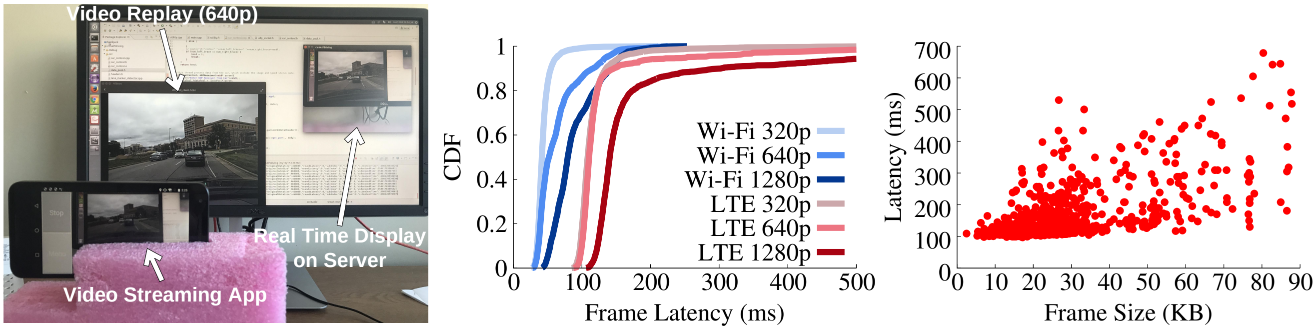
4 FEASIBILITY STUDY

In this section, we evaluate the feasibility of real-time video streaming over today's wireless networks used by vehicles, i.e., LTE [17] and Wi-Fi [19]. We present a case study with three levels of resolutions, while choosing the best resolution and bitrate is an open question and further study is required.

4.1 Real Time Streaming

As illustrated in Fig. 3a, we use a customized Android app to compress the raw video frames (in YUV420 format) and send to remote server in real time. The video is compressed by using video compression standard H.264 or MPEG-4 [13]. In H.264, there are two types of frames, I-frame and P-frame. An I-frame, or Intra-coded picture, is a complete image. P-frame (Predicted picture) hold only the part that changes between frames. In other non-real time applications, it can also use both previous and forward frames to generate B-frame (Bidirectional predicted picture) to better compress the frame.

We borrow video parameter settings from popular VoIP applications, i.e., Skype [26] and Google Hangout [24]. In our setup, we conduct case study with three resolutions of 320x240, 640x480 and 1280x960, with bitrate 0.5Mbps, 1Mbps and 4Mbps, respectively. We use UDP to send the compressed frames with a frame rate of 10 over LTE and Wi-Fi networks. The I-frame interval is 1 second, i.e., there is one I-frame every 10 frames. The compression ratio ranges from 5% to 15% for different frames. The server decompresses the video frame by using GStreamer pipeline [5] and sends back the timestamp associated with the frame. The Android app records the two-way latency and frame sizes into sqlite database.



(a) Real time streaming. (b) The two-way latency of video frames. (c) Latency of different frame sizes.

Figure 3: Latency measurement of real-time streaming in Wi-Fi and LTE networks. Frame loss rate is from 0.5% to 2% in different networks and resolutions (and bitrates) settings.

4.2 Video Frame Latency

We measure the two-way latency of the compressed video frames in both LTE and Wi-Fi networks. The measurements in Wi-Fi network is used to compare with LTE for protocol overhead. The road experiments are conducted in LTE networks. In both networks, we use the same remote UDP server with a global IP address. For the measurements in LTE networks, we fix the Android phone (Nexus 5X) in vehicle mount holder to record front views. We drive the vehicle for three 5min trips to record two-way frame latency, frame size and loss rate of different resolutions. For the measurements in Wi-Fi network, we use the Android phone to stream a prerecorded video through a single 802.11n Wi-Fi access point. The server is 3 miles away from the streaming location. The two-way latency in both networks is shown in Fig. 3b. The median latency in LTE and Wi-Fi networks are around 100ms and 50ms, respectively. One reason for higher latency in LTE networks is the protocol overhead. Since we use only one Wi-Fi access point, the overhead of handoff is excluded. With carefully designed handoff and scheduling algorithm [19], we believe Wi-Fi is a good candidate for vehicle connectivity in urban area.

4.3 Frame Size and Optimization

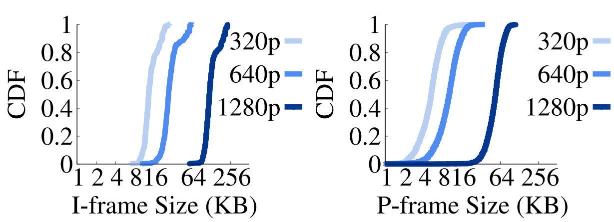


Figure 4: The size compressed frames, with compression ratio 5%-20% of original YUV420 format.

The frame size distributions of I-frames and P-frames are illustrated in Fig. 4. The median size of I-frames is 2-4 times larger than that of P-frames. This is because the video encoder fails to predict the next frame and the difference between actual frame and predicted frame is much higher than expected. We observe that large frame size may cause long tail latency, as illustrated in Fig. 3c. Some optimization techniques can be used to further reduce the size of the frames. The I-frame interval can be adjusted according to vehicle dynamics. At high speed or during turns, the I-frame interval can be increased as the it is hard to predict next frame in such cases. When the vehicle is waiting at red light, the frame rate or resolution can be reduced since the human operator cannot drive the vehicle anyway. Further, a 3D map of road and infrastructures can be built and cached at both the self-driving system and the server, only the differences (i.e., dynamic objects such as vehicles and pedestrians) are transmitted to the server. Such optimizations can reduce the peak size of the frames to potentially avoid long tail latency.

5 RELATED WORK

5.1 Self-Driving Systems

Corporations like Waymo, Mercedes-Benz and AutoX are trying to develop fully self-driving vehicles [3, 14, 22]. Waymo uses LIDAR as the primary input for object detection [22]. AutoX proposes camera-first self-driving solution to reduce the cost to build a self-driving vehicle [3]. [4] presents a sensory-fusion perception framework that combines LIDAR point cloud and RGB images as input and predicts oriented 3D bounding boxes. [9] describes the architecture and implementation of an autonomous vehicle designed to navigate using locally perceived information in preference to potentially inaccurate or incomplete map data. [7] presents networked self-driving vehicles to coordinate and form an edge computing platform. We believe remote control system can act the safe backup for such self-driving systems.

5.2 Low Latency Networks

Reducing network latency is an active area of research. [18] investigates the causes of latency inflation in the Internet and proposes a grand challenge for the networking research community: a speed-of-light Internet. [1, 25] propose various architectures and techniques for high capacity and low latency 5th generation mobile networks. [20] discusses the requirements of system design for real-time streaming. [19] presents a Wi-Fi based roadside hotspot network to operate at vehicular speeds with meter-sized picocells. [8] uses speculation to predict future frames to reduce latency for mobile cloud gaming. [10] measures the performance of Skype over today's LTE networks and illustrate the inefficiencies of Skype protocols. [6] develops a passive measurement tool to study the inefficiency in today's LTE networks. [2] presents the features to improve quality of service in LTE networks. [21] presents the inefficiencies of current VoLTE architectures. All these work can inspire the design of remote control systems for self-driving vehicles.

6 DISCUSSION

6.1 Multiple Camera Feeds

We evaluate only single camera feed, while it is necessary to stream multiple camera feeds from both the vehilce itself and surrounding vehicles. Streaming multiple camera feeds will increase the data volume and further optimization techniques are required.

6.2 Video Resolution, Bitrate and Quality

In this paper, we present a case study with three levels of resolutions, while choosing the best resolution and bitrate for this application is still an open question. Different video bitrates provide different video qualities for various resolutions. The question is what is the minimum requirements and parameter settings to provide enough quality for remote operator to be fully aware of the surrounding situations. We expect that a live streaming protocol should adjust video bitrate according to network bandwidth and provide the best-effort video quality.

7 CONCLUSION

We propose to use remote control when the self-driving system fails to understand the environment or cannot match the road information with predefined traffic rules. It raises many open questions to design a remote control system and infrastructure. We present case studies in this work and advocate further research into the challenging issues for augmenting self-driving with remote control.

ACKNOWLEDGMENTS

We would like to acknowledge the anonymous reviewers and our shepherd Dr. Giovanni Pau for his detailed comments, guidance and inspiration. This research project is supported in part by the US National Science Foundation through awards CNS-1345293, CNS-14055667, CNS-1525586, CNS-1555426, CNS-1629833, CNS-1647152 and CNS-1719336.

REFERENCES

[1] Mamta Agiwal, Abhishek Roy, and Navrati Saxena. 2016. Next generation 5G wireless networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials* (2016).

[2] Najah Abu Ali, Abd-Elhamid M Taha, and Hossam S Hassanein. 2013. Quality of service in 3GPP R12 LTE-advanced. *IEEE Communications Magazine* (2013).

[3] AutoX. 2016. Camera-first AI brings self-driving cars out of the lab and into the real world. https://www.autox.ai/. (2016).

[4] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. 2017. Multi-View 3D Object Detection Network for Autonomous Driving. In *IEEE CVPR*.

[5] GStreamer. [n. d.]. https://gstreamer.freedesktop.org/. ([n. d.]).

[6] Junxian Huang, Feng Qian, Yihua Guo, Yuanyuan Zhou, Qiang Xu, Z Morley Mao, Subhabrata Sen, and Oliver Spatscheck. 2013. An in-depth study of LTE: effect of network protocol and application behavior on performance. In *ACM SIGCOMM*. ACM.

[7] Eun-Kyu Lee, Mario Gerla, Giovanni Pau, Uichin Lee, and Jae-Han Lim. 2016. Internet of Vehicles: From intelligent grid to autonomous cars and vehicular fogs. *International Journal of Distributed Sensor Networks* (2016).

[8] Kyungmin Lee, David Chu, Eduardo Cuervo, Johannes Kopf, Yury Degtyarev, Sergey Grizan, Alec Wolman, and Jason Flinn. 2015. Outatime: Using speculation to enable low-latency continuous interaction for mobile cloud gaming. In *MobiSys*. ACM.

[9] John Leonard, Jonathan How, Seth Teller, Mitch Berger, Stefan Campbell, Gaston Fiore, Luke Fletcher, Emilio Frazzoli, Albert Huang, Sertac Karaman, et al. 2008. A perception-driven autonomous urban vehicle. *Journal of Field Robotics* (2008).

[10] Li Li, Ke Xu, Dan Wang, Chunyi Peng, Kai Zheng, Haiyang Wang, Rashid Mijumbi, and Xiangxiang Wang. 2017. A measurement study on Skype voice and video calls in LTE networks on high speed rails. In *IWQoS*. IEEE.

[11] Todd Litman. 2014. Autonomous vehicle implementation predictions. *Report* (2014).

[12] Daily Mail. [n. d.]. http://www.dailymail.co.uk/sciencetech/article-4889594/Google-s-spent-1-1-BILLION-self-driving-cars.html. ([n. d.]).

[13] Detlev Marpe, Thomas Wiegand, and Gary J Sullivan. 2006. The H. 264/MPEG4 advanced video coding standard and its applications. *IEEE communications magazine* (2006).

[14] Autonomous Mercedes-Benz Research & Development. [n. d.]. http://mbrdna.com/divisions/autonomous-driving/. ([n. d.]).

[15] Lothar Pantel and Lars C Wolf. 2002. On the impact of delay on real-time multiplayer games. In *Proceedings of the 12th international workshop on Network and operating systems support for digital audio and video*. ACM, 23–29.

[16] Scott Drew Pendleton, Hans Andersen, Xinxin Du, Xiaotong Shen, Malika Meghjani, You Hong Eng, Daniela Rus, and Marcelo H Ang. 2017. Perception, Planning, Control, and Coordination for Autonomous Vehicles. *Machines* (2017).

[17] Hanbyul Seo, Ki-Dong Lee, Shinpei Yasukawa, Ying Peng, and Philippe Sartori. 2016. LTE evolution for vehicle-to-everything services. *IEEE Communications Magazine* (2016).

[18] Ankit Singla, Balakrishnan Chandrasekaran, P Godfrey, and Bruce Maggs. 2014. The internet at the speed of light. In *HotNets*. ACM.

[19] Zhenyu Song, Longfei Shangguan, and Kyle Jamieson. 2017. Wi-Fi Goes to Town: Rapid Picocell Switching for WirelessTransit Networks. In *ACM SIGCOMM Computer Communication Review*. ACM.

[20] Michael Stonebraker, Uur Çetintemel, and Stan Zdonik. 2005. The 8 requirements of real-time stream processing. *ACM SIGMOD* (2005).

[21] Guan-Hua Tu, Chi-Yu Li, Chunyi Peng, Zengwen Yuan, Yuanjie Li, Xiaohu Zhao, and Songwu Lu. 2016. VoLTE*: A Lightweight Voice Solution to 4G LTE Networks. In *HotMobile*. ACM.

[22] Waymo. 2016. On the road to fully self-driving, Waymo safety report. https://storage.googleapis.com/sdc-prod/v1/safety-report/waymo-safety-report-2017-10.pdf. (2016).

[23] Wikipedia. [n. d.]. https://en.wikipedia.org/wiki/Autonomous_car. ([n. d.]).

[24] Chenguang Yu, Yang Xu, Bo Liu, and Yong Liu. 2014. Can you SEE me now? A measurement study of mobile video calls. In *INFOCOM, 2014 Proceedings IEEE*. IEEE.

[25] Shunqing Zhang, Xiuqiang Xu, Yiqun Wu, and Lei Lu. 2014. 5G: Towards energy-efficient, low-latency and high-reliable communications networks. In *ICCS*. IEEE.

[26] Xinggong Zhang, Yang Xu, Hao Hu, Yong Liu, Zongming Guo, and Yao Wang. 2012. Profiling skype video calls: Rate control and video quality. In *INFOCOM, 2012 Proceedings IEEE*. IEEE.