

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

185,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Visibility-Based Technologies and Methodologies for Autonomous Driving

Said Easa, Yang Ma, Ashraf Elshorbagy, Ahmed Shaker, Songnian Li and Shriniwas Arkatkar

Abstract

The three main elements of autonomous vehicles (AV) are orientation, visibility, and decision. This chapter presents an overview of the implementation of visibility-based technologies and methodologies. The chapter first presents two fundamental aspects that are necessary for understanding the main contents. The first aspect is highway geometric design as it relates to sight distance and highway alignment. The second aspect is mathematical basics, including coordinate transformation and visual space segmentation. Details on the Light Detection and Ranging (Lidar) system, which represents the ‘eye’ of the AV are presented. In particular, a new Lidar 3D mapping system, that can be operated on different platforms and modes for a new mapping scheme is described. The visibility methodologies include two types. Infrastructure visibility mainly addresses high-precision maps and sight obstacle detection. Traffic visibility (vehicles, pedestrians, and cyclists) addresses identification of critical positions and visibility estimation. Then, an overview of the decision element (path planning and intelligent car-following) for the movement of AV is presented. The chapter provides important information for researchers and therefore should help to advance road safety for autonomous vehicles.

Keywords: Lidar, traffic visibility, infrastructure visibility, high-precision maps, sight distance, highway design, technologies

1. Introduction

Autonomous vehicle (AV) will reduce human errors and is expected to lead to significant benefits in safety, mobility, and sustainability [1–5]. The technical feasibility of automated highways has been demonstrated in San Diego, California in 1997 [6, 7]. The technology is emerging around the world on both passenger and freight sides. Autonomous vehicles have already started to appear on the roads across the globe. Clearly, as the AV market expands, transportation professionals and researchers must address an array of challenges before AV soon becomes a reality. Several government and industry entities have deployed demonstrations and field tests of the technology. Centres for testing and education, products, and standards for AV have also been established. Currently, researchers, scientists, and engineers are investing significant resources to develop supporting technologies.

The self-driving system involves three main elements: orientation, visibility, and decision, as shown in **Figure 1**. For each element, certain functions are performed with the aid of one or a combination of technologies. For AV orientation, the position of the vehicle is determined mainly using Global Navigation Satellite System (GNSS). To increase reliability and accuracy, this technology is supplemented with other data gathered from specific perception technologies, such as cameras and internal measurement devices (tachometers, altimeters, and gyroscopes).

The visibility of AV, sometimes referred to as perception, involves infrastructure detection (e.g. sight obstacles, road markings, and traffic control devices) and traffic detection (other vehicles, pedestrians, and cyclists). High-precision (HP) maps help the AV not only perceive the environment in its vicinity, but also plan for the turns and intersection far beyond the sensors' horizons. The other main technology that represents the 'eye' of the AV is the Light Detection and Ranging (Lidar). Other technologies used for infrastructure visibility include video cameras which can detect traffic signals, read road signs, keep track of other vehicles, and record the presence of pedestrians and other obstacles. Radar sensors can determine the positions of other nearby vehicles, while ultrasonic sensors (normally mounted on wheels) can measure the position of close objects (e.g. curbs). Lidar sensors can also be used to identify road features, like lane markings. Three primary sensors (camera, radar, and Lidar) work together to provide AV with visuals of its 3D environment and help detect the speed and distance of nearby objects. The visibility information is essential for the safe operation of the AV. For example, the information can be used to ensure that adequate sight distance (SD) is available and if not to take appropriate actions.

The decision of AV is based on the data generated by the AV sensors and HP maps. Examples of the decisions that are made by AV include routing, changing lanes, overtaking vehicles, stopping at traffic lights, and turning at an intersection. The decision is made by an on-board centralized artificial intelligence (AI) computer that is linked with cloud data and include the needed algorithms and software.

For the AV to function effectively, it must communicate with the environment as part of a continuously updating, dynamic urban map [3]. This involves vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications. Internet of Things (IoT) technology provides telematics data about the condition and

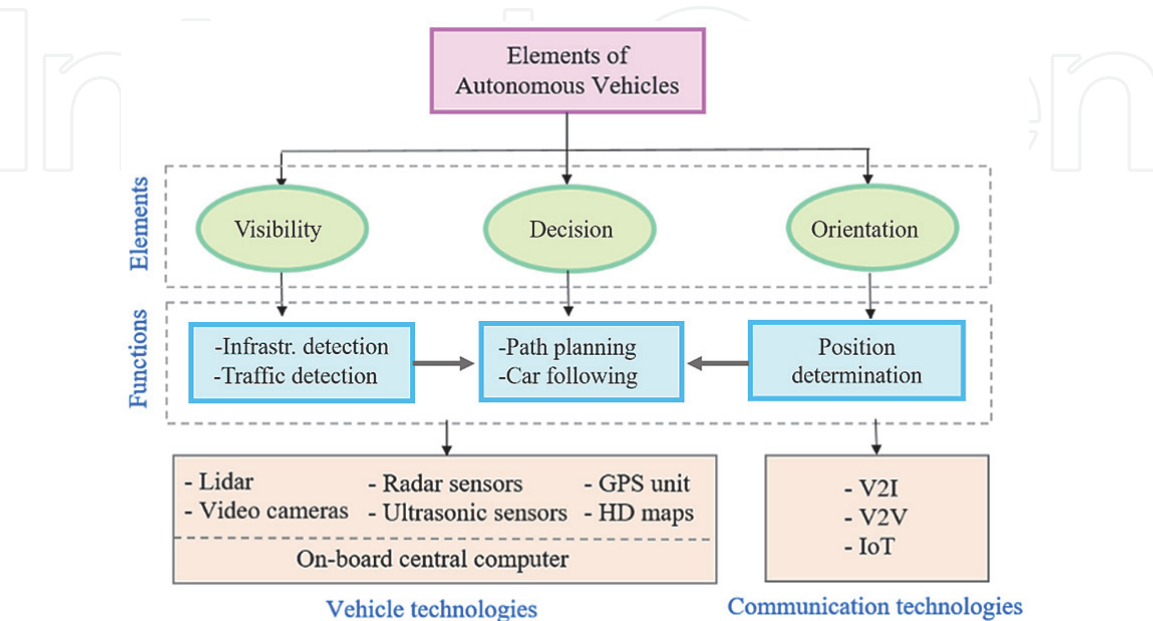


Figure 1.
Elements of autonomous vehicles and their functions and technologies.

performance of AV, such as real-time location and speed, idling duration, fuel consumption, and the condition of its drivetrain.

This chapter mainly focuses on the visibility of AV and briefly addresses the decision element. Section 2 presents important visibility fundamentals related to highway geometric design and mathematical tools. Section 3 presents details on the Lidar system, which represents the ‘eye’ of the AV. This section also introduces a new Lidar system that has great potential for high precision mapping for autonomous vehicles. Section 4 describes the methodologies related to infrastructure visibility and traffic visibility. Section 5 presents some implementation aspects and Section 6 presents the conclusions.

2. Visibility fundamentals

2.1 Highway sight distance

In the geometric design guides by American Association of State Highways and Transportation Officials and Transportation Association of Canada [8–10], the available sight distance for human-driven vehicles is measured from the driver’s height above the pavement surface. For autonomous vehicles, the driver’s eye is the Lidar. The ability of the AV to ‘see’ ahead is critical for the safe and efficient operation. Sufficient sight distance must be provided to allow the AV to stop, avoid obstacles on the roadway surface, overtake slow vehicles on two-lane highways, and make safe turns/crossings at intersections. For human-driven vehicles, the required sight distance is based on driver’s perception-reaction time (PRT), vehicle speed, and other factors. A value of 2.5 s is used for human-driven vehicles and a value of 0.5 s has been assumed for AV in the literature [11]. This smaller reaction time will result in shorter required sight distances for AV.

For autonomous vehicles, four basic types of SD are defined as follows:

- *Stopping Sight Distance (SSD)*: This is the distance traveled by the AV during system reaction time and the braking distance from the operating speed to stop.
- *Passing Sight Distance (PSD)*: This is the distance, including system reaction time, required by the AV on rural two-lane roads to allow the vehicle to pass a slower vehicle by using the opposing lane.
- *Decision Sight Distance (DSD)*: This is the distance that allows the AV to maneuver or change its operating speed or stop to avoid an obstacle on the roadway surface.
- *Intersection Sight Distance (ISD)*: This is the distance along a cross road with a right of way that must be clearly visible to the crossing AV so that it can decide and complete the maneuver without conflicting with the cross-road vehicles.

Since the AV response time is less than the driver’s PRT, the required SD for autonomous vehicles would be shorter than that for human-driven vehicles. The impact of AV on highway geometric design has been explored in a preliminary manner by making simple assumptions regarding system reaction time and Lidar field of view [11]. The study focused on SSD, DSD, and vertical curves.

The required SD for autonomous vehicles, Lidar height, and object height influence the design and evaluation of highway vertical, horizontal, and 3D alignments, as shown in **Figure 2**. The Lidar height above the pavement surface is critical in

determining the available sight distance. For crest vertical curves (**Figure 2a**), since the required SD for autonomous vehicles are shorter, the required Lidar height for safe operation, h_L , would be somewhat less than the design driver's eye height, h_1 . Thus, by placing the Lidar at or above h_L , the AV can safely operate on existing highways without the need for modifying their design. For a sag vertical curve with an overpass (**Figure 2b**), where the truck driver's eye height controls the traditional curve design, the required Lidar height for safe operation would be somewhat larger than the design driver's eye height. Thus, by placing the Lidar at or below h_L , the AV can safely operate on existing highways. For horizontal curves (**Figure 2c**), the Lidar height is not important for detecting horizontal obstacles, except when cut slopes are present.

For human-driven vehicles, sag vertical curves (**Figure 2d**) are designed based on the distance that the vehicle headlamp can illuminate ahead of the vehicle, where the angle of light beam upward from the vehicle plane is considered equal to 1° . What is interesting is that for AV the Lidar height is irrelevant to the operation on sag vertical curves. The reason is that Lidar detection does not require light and therefore an obstacle ahead of the autonomous vehicle can be detected under all light conditions. It is necessary, however, to ensure that the effective Lidar range is greater than the required SD.

Note that the geometry of SD shown in **Figure 2** represents individual horizontal or vertical alignments. For 3D alignment (combined horizontal and vertical curves), the AV can directly determine the available SD and compare it with the required SD. The sight line in this case may be obstructed by the pavement surface or by obstacles on the roadside, such as a building or a cut slope.

2.2 Mathematical tools

2.2.1 Overview of visibility modeling

Quantitative visibility estimation (VE) involves four basic components as shown in **Figure 3**: (1) sight point, (2) target points (e.g. lane marking, traffic sign, and stalled vehicle), (3) line of sight (LOS) which connects the sight point with target points, and (4) obstacle data that obstruct LOS (e.g. vegetation, barrier, and building). The purpose of VE is to determine whether the obstacles affect the visibility of the targets to the sight point in a mathematical manner.

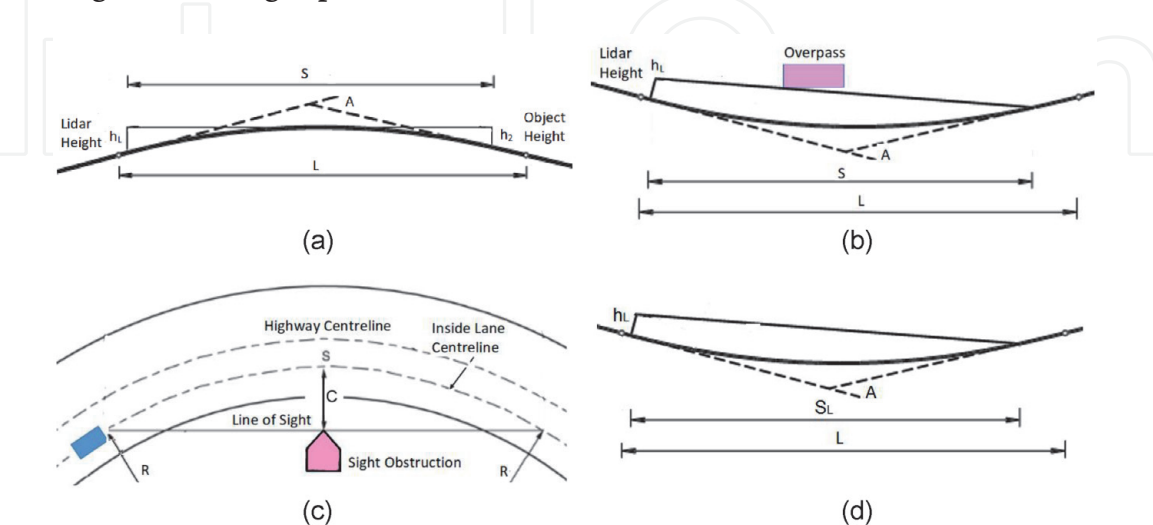


Figure 2. Effect of Lidar height and sight distance on the operation of AV on different highway alignments. Note: A = algebraic difference in grade, C = lateral clearance, h_L = Lidar height, h_2 = object height, S_L = effective Lidar range, L = vertical curve length, R = radius of horizontal curve, S = sight distance. (a) Crest vertical curve (b) sag vertical curve with overpass, (c) horizontal curve and (d) sag vertical curve.

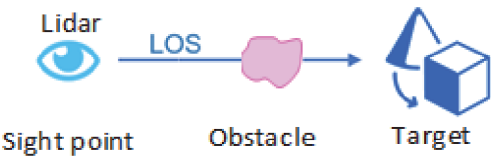


Figure 3.
Basic components of quantitative visibility estimation.

The accuracy of VE is closely associated with the precision of obstacle data. In the past, due to a lack of effective and affordable technique to collect dense and precise geospatial data, the obstacle data were conventionally represented by digital surface and terrain models generated from sparse geospatial points. However, it was proved that neither model could handle objects with complex shapes and may yield inaccurate VE results in some cases [12–14].

Over the past five years, mobile laser scanning (MLS) data have been recognized as a reliable data source for conducting visibility-related analyses [15–18]. MLS point clouds enable a very accurate and precise representation of real-world environment. In this case, a number of indoor computerized estimations on MLS point clouds are viable to replace risky and time-consuming outdoor field measurements [16]. In addition, MLS data are also the main data source for manufacturing HP maps which are indispensable to AV. Therefore, the mathematical model for estimating visibility in autonomous driving is built based on MLS data.

The general workflow of VE using MLS data is graphically presented in **Figure 4**. For a given eye position, both the target points and obstacle points undergo the same procedure (i.e. coordinate transformation and segmentation) and two respective depth maps of the same size are generated. The visibility estimation is achieved through comparing two 3D depth maps. The process of VE is described in detail next.

2.2.2 Coordinate transformation

To illustrate, consider VE at a single position. Let S be the sight point. Let \vec{F} be the forward direction. Let Φ and Ψ be the set of target points and obstacle points, respectively. To reduce the computational complexity, a local coordinate frame is established first as depicted in **Figure 5a**. Set S as the origin. Then, the Y' and X' axes are set along \vec{F} and the horizontal vector normal to \vec{F} , respectively. Finally, the Z' -axis is set along the upward direction perpendicular to the $X'SY'$ plane. The local coordinates of points (S , Φ and Ψ) are obtained via:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \mathfrak{R} \cdot \begin{bmatrix} x - x_s \\ y - y_s \\ z - z_s \end{bmatrix} = \begin{bmatrix} \cos \alpha \cdot \cos \beta & -\cos \beta \cdot \sin \alpha & \sin \beta \\ \sin \alpha & \cos \alpha & 0 \\ -\cos \alpha \cdot \sin \beta & \sin \alpha \cdot \sin \beta & \cos \beta \end{bmatrix} \cdot \begin{bmatrix} x - x_s \\ y - y_s \\ z - z_s \end{bmatrix} \quad (1)$$

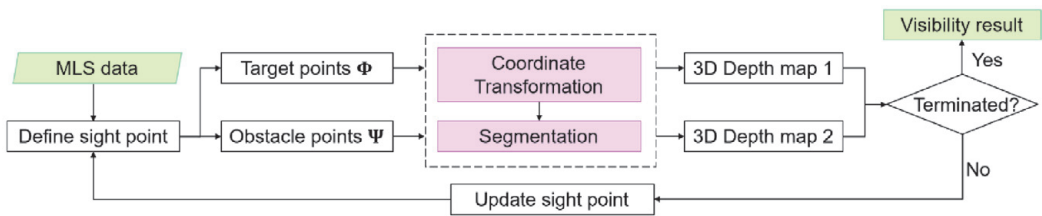


Figure 4.
Workflow of VE using MLS data.

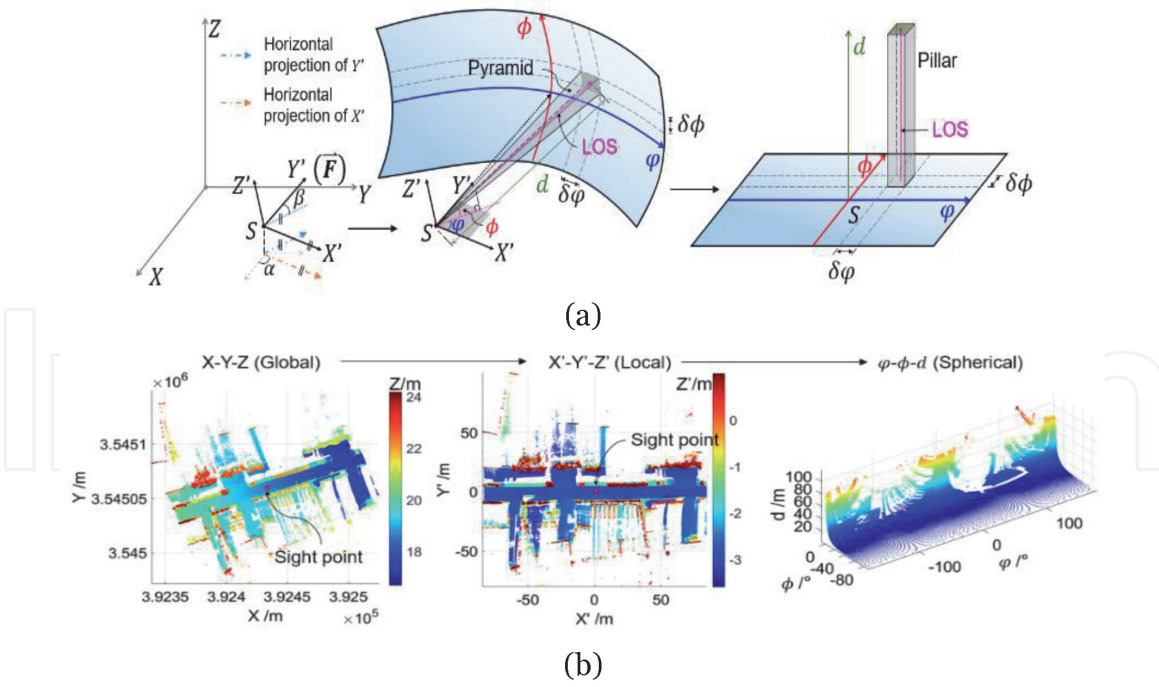


Figure 5. Coordinate transformation: (a) different coordinate systems and (b) visualization of points in different frames (generated with Matlab 2020b; the same for **Figures 6, 7, and 8-10**).

where \mathfrak{R} = the rotation matrix, $(x_s, y_s, z_s)^T$ = geodetic coordinates of the sight point, $(x, y, z)^T$ and $(x', y', z')^T$ = geodetic and local coordinates of points (S , Φ and Ψ), respectively, and α, β = rotation angles around the Z and X -axes, respectively.

To enable a more intuitive and efficient estimation of target visibility, the local Cartesian coordinates are further converted into spherical coordinates as follows,

$$\begin{cases} \varphi = \arctan \frac{y'}{x'} \\ \phi = \arctan \frac{z'}{\sqrt{x'^2 + y'^2}} \\ d = \sqrt{x'^2 + y'^2 + z'^2} \end{cases} \quad (2)$$

where $(\varphi, \phi, d)^T$ = azimuth, elevation, and radius of the spherical coordinates shown in **Figure 5a**.

Points in different coordinate frames are visualized in **Figure 5b**. The sight point marked in figure corresponds to S in **Figure 5a**. In the $\varphi-\phi-d$ space, φ and ϕ refer to the horizontal and vertical angles between a LOS and the forward direction, respectively, while d measures the depth from the sight to the target point. Conventionally, as shown in **Figure 5a**, a sight cone or sight pyramid is constructed around a given LOS in a local frame to detect whether the obstacle points therein touch that LOS [12, 17]. However, when the number of target points is very large, the practice of building sight pyramids and then searching for points therein in dense MLS point clouds is quite time-consuming. With the coordinate transformation, a sight pyramid in the $x'-y'-z'$ space is equivalent to a pillar in the $\varphi-\phi-d$ space. In this case, it is reasonable to assume that when $\delta\varphi$ and $\delta\phi$ are small, only the point with the minimum d can be seen by an observer in a pillar. This process is more straightforward than detecting the closest point in a sight pyramid in the $x'-y'-z'$ coordinate system.

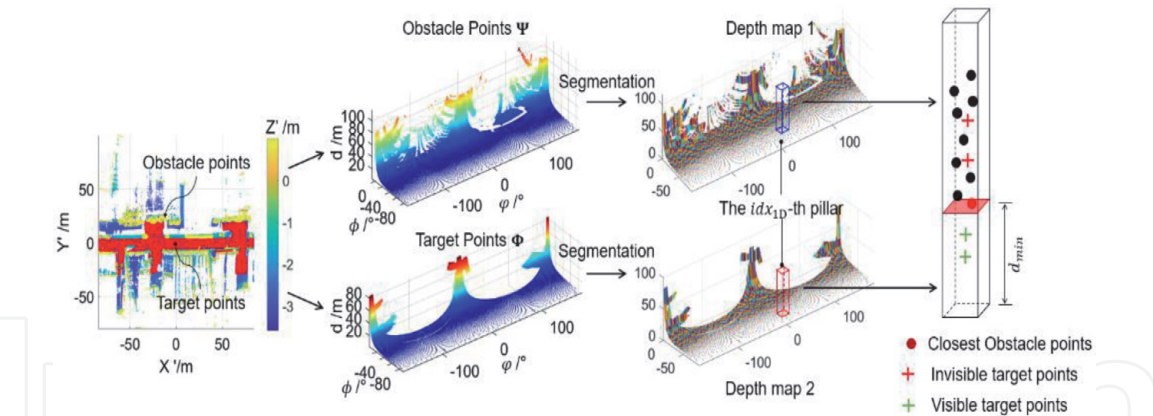


Figure 6.
 Comparison of two depth maps.

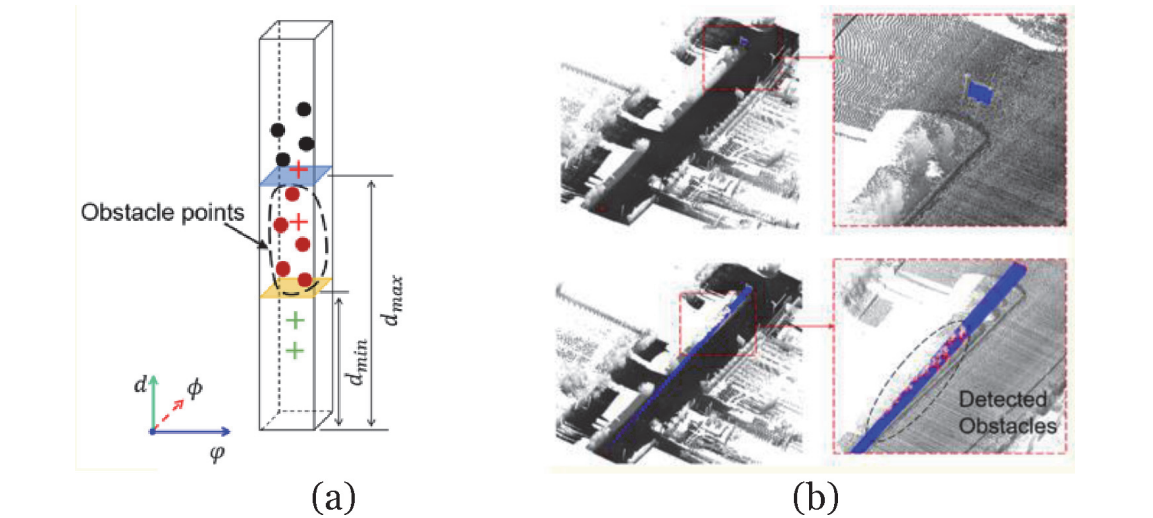


Figure 7.
 Sight obstacle detection: (a) principle and (b) results.

2.2.3 Visual space segmentation

Based on the preceding assumption, the main task is to segment points in the φ - ϕ - d space into different pillars. To reduce the computational complexity and thus improve the efficiency, a linear index based segmentation procedure was developed and applied to partition points. The procedure assigns a new one-dimensional (1D) index to each point and can divide point cloud data into pillars or voxels. Taking the voxel generation for illustration, the segmentation procedure comprises three main steps as follows:

Step 1: Grid the original 3D data $(a, b, c)^T$ using the following formula:

$$\begin{cases} a_g = \left\lfloor \frac{a}{\delta_a} \right\rfloor \cdot \delta_a \\ b_g = \left\lfloor \frac{b}{\delta_b} \right\rfloor \cdot \delta_b \\ c_g = \left\lfloor \frac{c}{\delta_c} \right\rfloor \cdot \delta_c \end{cases} \tag{3}$$

where $\lfloor \cdot \rfloor$ = a function that rounds the number to an integer, $\delta_a, \delta_b, \delta_c$ = user-defined voxel dimensions along the a, b , and c axes, respectively. **Figure 11a** shows the dense points that are converted to the gridded data.

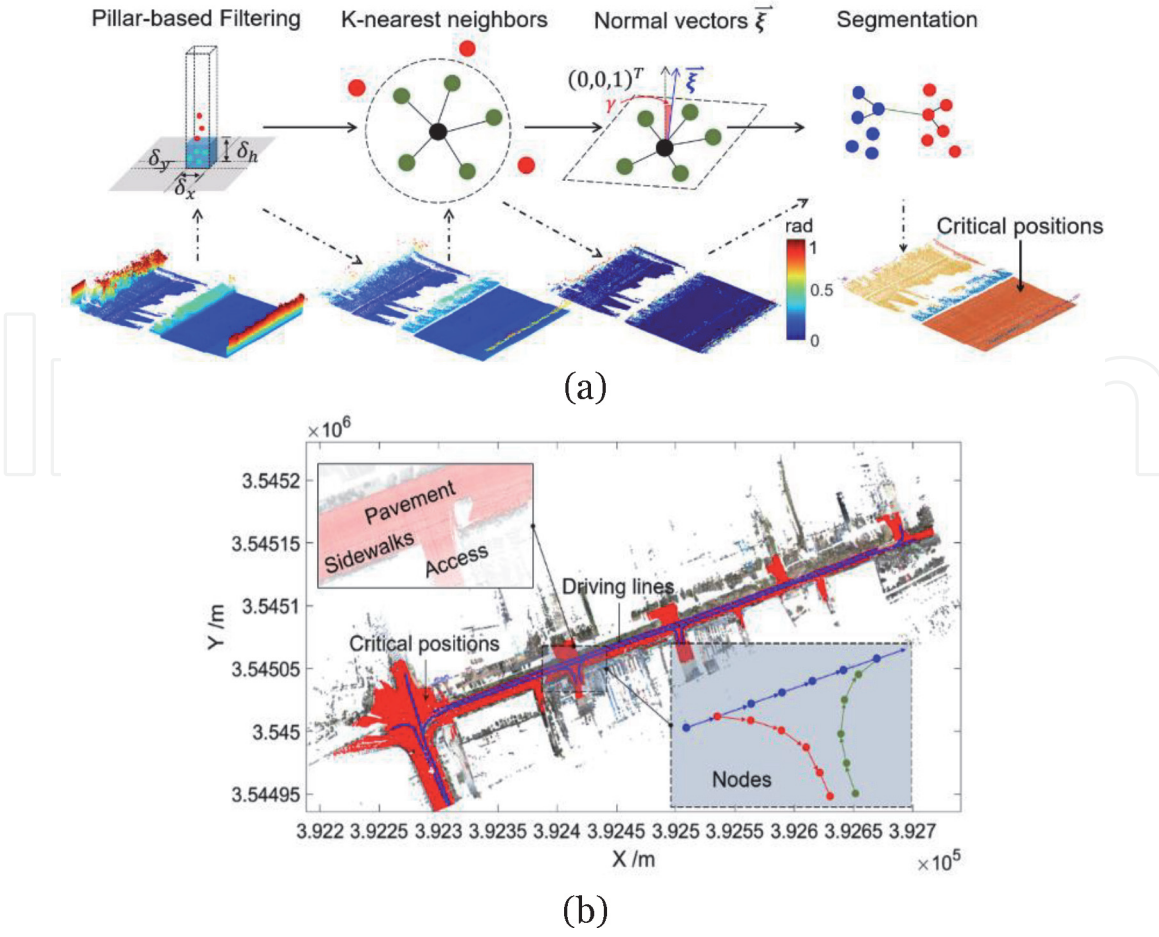


Figure 8. Extraction of critical positions: (a) general process of identifying planar ground points and (b) critical positions and driving lines.

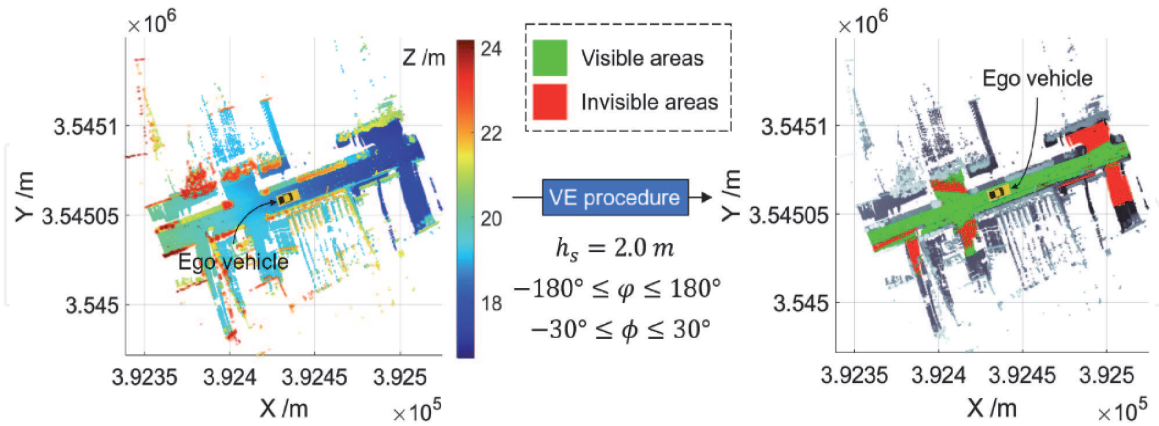


Figure 9. VE results at a single node.

Step 2: Let D_a, D_b , and D_c be the distances along the a, b , and c axes from $(\min a_g, \min b_g, \max c_g)^T$ to $(\max a_g, \max b_g, \min c_g)^T$, respectively. Let d_a, d_b , and d_c be the distances along the a, b , and c axes from $(\min a_g, \min b_g, \max c_g)^T$ to a certain grid point P , respectively. A null $(D_b + 1) \times (D_a + 1) \times (D_c + 1)$ cell matrix $\{\Theta\}$ and the same-size numerical matrix $[\Lambda]$ are then constructed. The location of the grid point P in $\{\Theta\}$ or $[\Lambda]$ is $(d_b + 1, d_a + 1, d_c + 1)^T$. The location can be converted to a linear index idx_{1D} as follows

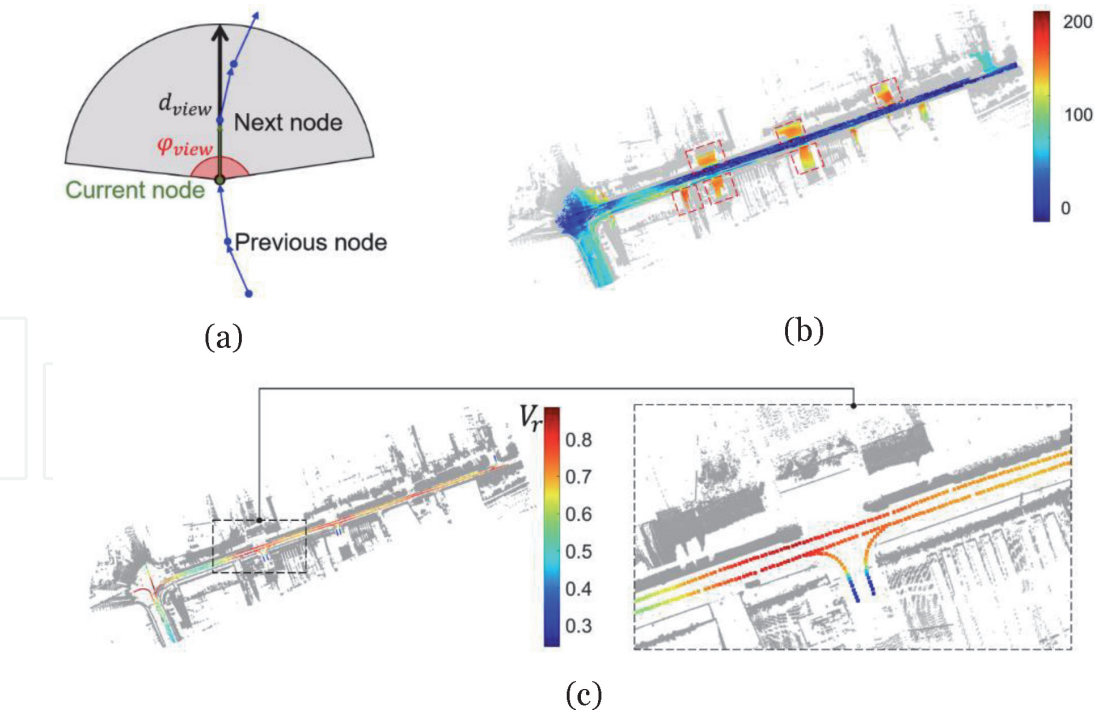


Figure 10.
Visualization of VE results: (a) VE process, (b) cumulative times of invisibility, and (c) visibility ratio.

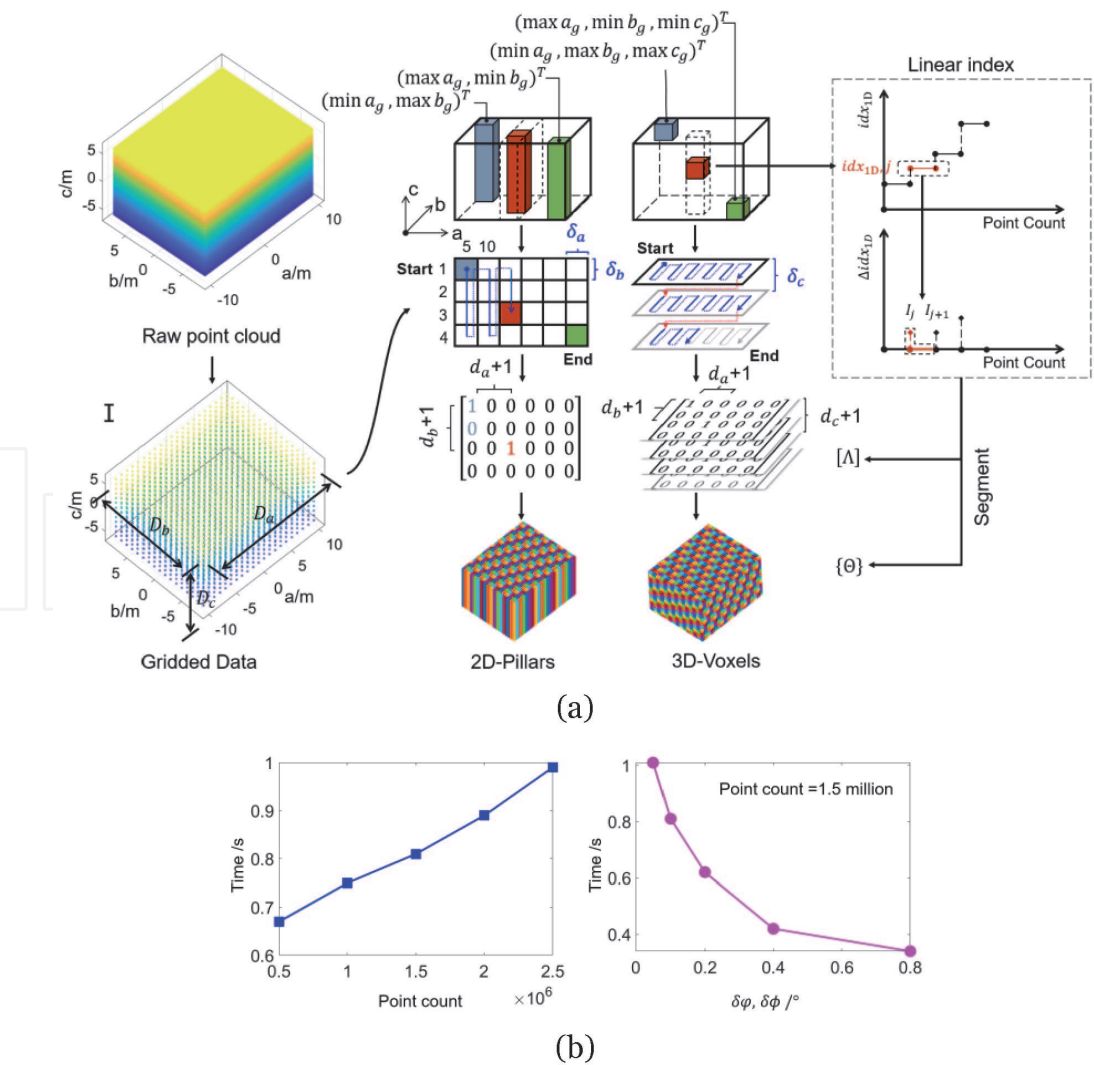


Figure 11.
Linear index-based segmentation: (a) process and (b) time performance.

$$idx_{1D} = d_c \cdot (D_b + 1) \cdot (D_a + 1) + d_a \cdot (D_b + 1) + d_b + 1 \quad (4)$$

where $d_a \leq D_a, d_b \leq D_b, d_c \leq D_c$ and $idx_{1D} \leq (D_b + 1) \cdot (D_a + 1) \cdot (D_c + 1)$.

In a matrix, each element can be accessed through either its 3D location or the linear index idx_{1D} [19–21]. Using Eq. (4), a 1D index is obtained for each point.

Step 3: Sort the points by increasing idx_{1D} , where the points falling inside a voxel will share the same idx_{1D} , as shown in **Figure 11a**. By computing the difference Δidx_{1D} between two consecutive idx_{1D} , the points in a certain voxel is determined by detecting the positions where Δidx_{1D} jumps. Suppose

$I\{I_1, I_2 \dots I_j \dots I_m | j > 1, m \leq (D_b + 1) \cdot (D_a + 1) \cdot (D_c + 1) - 1\}$ is the set of indices of $\Delta idx_{1D} > 0$. The I_j -th to I_{j+1} -th points are placed in the $idx_{1D,j}$ -th element of $\{\Theta\}$. In the meantime, set the $idx_{1D,j}$ -th element of $[\Lambda]$ equal to 1. Regarding the pillars generation, the 1D index is written as

$$idx_{1D} = d_a \cdot (D_b + 1) + d_b + 1 \quad (5)$$

Due to the high computational simplicity, detecting the jumps of Δidx_{1D} to segment 3D data is faster than using the exhaustive searching or the kd-tree neighbors searching algorithm. Besides, a corresponding binary matrix is generated in the process of segmentation, which may aid in detecting either the indices of non-empty pillars or the connected components (CC) when necessary.

Figure 11b shows the time performance of the linear index based segmentation procedure on a computer with the processor of Intel®371 Core™ i5-6600, RAM of 16 GB and GPU of Nvidia®GTX-960. As $\delta\varphi$ and $\delta\phi$ are both set as 0.1° , the time the segmentation procedure takes is linearly correlated with number of points. Notably, it can process 2 million points within 1 seconds (s). Given the same dataset, the processing time of the indexing operation is negatively and exponentially correlated with the angular resolutions. However, the efficiency is still satisfactory as it takes less than 1 s to handle 1.5 million points when both $\delta\varphi$ and $\delta\phi$ are set as 0.05° .

Using the procedure shown in **Figure 11a**, the target points Φ and obstacle points Ψ in the $\varphi - \phi - d$ space can be separately partitioned into numerous pillars and thus two 3D depth maps are generated as shown in **Figure 6**. It is noteworthy that the starting and the terminating points (see **Figure 11**) for calculating the linear indices of Φ and Ψ are identical (i.e. the size of generated $[\Lambda]$ or $\{\Theta\}$ is the same). As such, there is a one-to-one correspondence between pillars of depth maps 1 and 2. The binary matrix $[\Lambda]$ of depth map 2 can help determine the positions or indices of pillars that have target points (i.e. $[\Lambda] == 1$). Then the obstacle points in the corresponding pillar of depth map 1 can be retrieved via the linear index and compared with the target points as noted in **Figure 6**. Let d_{min} bet the minimum distance of the closest obstacle point to the sight point. As previously mentioned in Section 2.2.2, when $\delta\varphi$ and $\delta\phi$ are small, the target points are invisible (marked in red in Figure) if their d -values exceed d_{min} , otherwise they are visible (marked in green in Figure).

3. Visibility equipment

3.1 Overview

A growing number of diverse end users requires 3D data as many new outdoor and indoor application domains benefits from three dimensional (3D) maps. The new application domains include AV, smart cities, asset management, augmented virtual reality, as-built drawings, and even gaming industry. Active and passive

sensors are the main sensor types that can be used for scanning the surrounding environment and generating 3D data. Lidar scanners are the main active sensors, while optical cameras are the main passive sensors. The Lidar sensors are considered the standard sensors used for 3D scanning and mapping. Unlike passive sensors, Lidar scanners do not require an external source of light to illuminate the target. Thus, it can scan day or night under all lighting conditions and with a higher resiliency to scan in adverse weather conditions. The technological advances in Lidar scanners and its miniaturization are rapidly growing and can be deployed for several transportation application domains, such as AV, road furniture mapping, road condition assessment, and 3D visualization.

3.2 Lidar operation

A Lidar scanner is an active Remote Sensing Sensor (RSS) that uses light as the source of target illumination [22]. The Lidar unit emits a pulsed light beam or a continuous light wave that hits the object and reflects back to the sensor. The precise measurement of the range from the sensor to the target might follow one of two methods. The first method involves the accurate measurement of the Time of Flight, which is the time interval that has elapsed between the emission of a short (but intense) light pulse by the sensor and its return after being reflected from an object to the sensor. Then, the range is given by

$$R = v t / 2 \quad (6)$$

where R = range (m), v = speed of light (m/s), and t = time interval measured from the emission to reception of the light pulse (s), where $v = \sim 299,792,458$ m/s.

The precision of the time measurement determines the precision of the range measurement which is given by

$$\Delta R = t \Delta v / 2 + v \Delta t / 2 \quad (7)$$

where ΔR = range precision, Δv = speed of light precision, and Δt = time interval precision. This method is commonly used in most Lidar systems.

The precision of the time measurement determines the precision of the range measurement, with the high-end Lidar scanners (surveying scanners) capable of measuring single pulse timing to a 3 picosecond (ps) accuracy and thus achieving a 1 mm range resolution. The range precision is given by

$$R = (M\lambda + \Delta\lambda) / 2 \quad (8)$$

where M = integer number of wavelengths (λ), and $\Delta\lambda$ = fractional part of the wavelength.

Generally, laser scanners survey the surrounding environment by steering the light beam through a mirror or a prism mechanism to cover one direction (e.g. vertical direction). To provide a sequence of profiles around the vertical axis of the laser unit and generate a 3D point cloud of the area around the laser unit, a controlled and measured motion in another direction (e.g. azimuth direction in the static terrestrial laser scanning case) is applied. Nevertheless, if the laser unit is mounted on a moving platform, the controlled and measured motion in the azimuth direction may be substituted by the platform movement.

Beam divergence is yet another factor that affects the 3D point cloud generation [22]. The light beam is collimated when emitted from the laser unit, but as the light beam propagates, the beam radius or diameter increases and the increase is related

to the distance the beam travels. Beam divergence is an angular measure that relates this increase to the distance it travels. The divergence affects the footprint that is measured by the beam. Thus, the measured distance represents a wider area on the target and, in turn, will decrease the specificity of the measured distance, as it will miss any position variation within the footprint. The effect is further highlighted in the case of long-range sensors, such as those used in Airborne Laser Scanning (ALS). This explains the narrow beam divergence needed in the sensors used for ALS, which is typically 0.5 mrad or less.

The Lidar sensor can be used in a mobile or static mode. In the mobile mode, the sensor is mounted on a moving platform such as AV or included as part of a 3D mapping system for mobile mapping and 3D map generation. In the static mode, also known as roadside Lidar for traffic monitoring, the sensor is mounted at a fixed location at an intersection to measure real-time traffic volume and speed. In the static mode, the sensor can use power over Ethernet as a versatile way of providing power to the sensor. The sensed data can then be classified using AI and big data algorithms as pedestrian/cyclist/vehicles objects, where a unique ID is then assigned to each classified object. The object can be continuously tracked using the position, direction, and speed. A complete solution that empowers a smart traffic flow management can be designed by integrating the Lidar within a perception software and IoT communications, thus improving mobility.

3.3 Types of Lidar scanners

Spinning multi-beam Lidar scanners, which are relatively recent, have been introduced to meet AV industry requirements. Unlike 2D laser scanners that depend on the platform movement which covers the third dimension, the sensors have multiple beams with pairs of an emitter and a receiver for each beam. Each beam is oriented at a fixed vertical angle from the sensor origin [23]. The multi-beam mechanism spins mechanically around a spinning axis to cover 360° horizontal field of view. The frequency of the spinning axis frequency of the Velodyne Lidar sensors reaches 20 Hz, thus enabling a fast and rich 3D point cloud of the vehicle's environment and enhancing its 3D perception. Three examples of these sensors are shown in **Figure 12**.

The solid-state flash Lidar is yet another Lidar technology. This technology illuminates large areas simultaneously and measures the reflected energy on a photonic phased array in an analogy of the digital camera complementary metal-oxide-semiconductor sensor. Unlike the previously-discussed sensor data measurement mechanisms, solid-state Lidar sensors do not have any moving parts and the miniaturization allows on-chip lasers [23]. Three examples of these sensors are shown in **Figure 13**.

It is worth noting that the current rapid increase in laser sensor technology is being driven by the AV application domain. The solid-state Lidar sensors are very



Figure 12. Examples of spinning multi-beam laser sensors: (a) Velodyne VLP-32c, (b) Hesai Pandar64, and (c) Quanergy M8. Source: (a) www.velodyne.com (b) www.hesai.tech, and (c) www.quanergy.com.

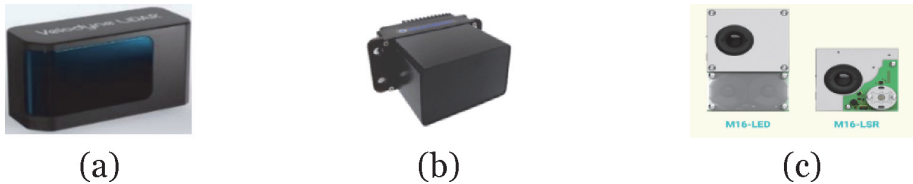


Figure 13.
Examples of solid state Lidar: (a) Velodyne Velarray; (b) Quanergy S3, and (c) Leddartech M16-LSR. Source: (a) www.velodyne.com, (b) www.quanergy.com, and (c) www.leddartech.com.

promising as they are specifically designed for vehicle environment grid occupancy detection and collision avoidance. As the technology advances, however, it is anticipated that the solid-state Lidar technology will have a substantially positive effect on the 3D mapping field. Two Livox Lidar sensors are available: Mid-70 which has zero blind spot and Avia which has a detection range of up to 450 m along with the multi-scanning modes (repetitive and non-repetitive). These sensors better meet the needs for low-speed autonomous driving (Mid-70 sensor) and such applications as topographic surveying and mapping and power line inspection (Avia sensor). The sensors are shown in **Figure 14**.

3.4 New Lidar system

3D maps are a key infrastructure component needed for smart cities and smart transportation applications. Mobile mapping allows the generation of 3D maps for very large areas, which are considered cost-prohibitive to be mapped using conventional mapping equipment. Active sensors are the primary RSS in a number of mobile mapping systems. A mobile mapping system (MMS) is a moving platform (typically a vehicle) where the Lidar-based system is mounted on. This setup allows capturing data of large areas quickly, which is needed if conventional terrestrial mapping is to be used. Mapping-grade MMS normally achieves sub-meter accuracy, while survey-grade MMS achieves cm-level accuracy, where cost is typically US \$400 k and US\$1 M, respectively.

Several factors hinder the current mobile mapping systems usage for many user segments, including major investment, operational cost, unease of deployment, and required level of expertise. The design, development, and implementation of a new Lidar-based generic 3D mapping system has been carried out at the Department of Civil Engineering, Ryerson University [24, 25]. The developed system uses the relatively low-cost recently released RSS. The optimized selection of the sensors, the smart integration of the 3D mapping system components on both the hardware and software allow a higher level of versatility, ease of deployment, and substantial cost reduction, compared to commercially systems, while maintaining a comparable accuracy metrics. The system can be used on drones, cars, or as a stationary device. The characteristics of the new system are shown in **Table 1**.



Figure 14.
Examples of the Livox Lidar sensors: (a) Livox Mid-70 and (b) Livox Avia. Source: www.livoxtech.com.

Feature	Value
System accuracy	5 cm @ 50 m (1 σ)
Weight (battery included)	1.5 kg
Diameter	10.3 cm
Height	15 cm
HFOV (VFOV)	360° (+15° to -15°)
Data	300,000 pts./sec (single return) 600,000 pts./sec (dual return)
Power consumption	19 W (Autonomy ~1.2 hr)
Operating Temperature:	-10°C to +60°C

Table 1.
Features of new Lidar 3D mapping system.

A sample of the 3D point cloud for a block at Ryerson University campus collected by the new system and colorized by height is shown in **Figure 15**. The figure clearly shows thin features such as overhead electricity wires, the traveled way, sidewalk, pedestrian crosswalk, traffic lights, and other vehicles.

The new Lidar-based system holds great potential for implementation in 3D map generation for AV. The drive behind the Lidar sensor technology advancement is to build a very low-cost, miniaturized sensor that can provide the AV with the ability of a robust 3D perception of the environment. A number of different competing factors need to be optimized, including sensor characteristics, such as range, range precision, horizontal and vertical fields of view, and size and power consumption. The main objective is to allow obstacle detection in a fast and reliable manner even at long ranges, thus ensuring safe vehicle operation.

3.5 Potential applications

The new Lidar sensors can benefit a multitude of application domains within the transportation sector. Those sensors are used to create the 3D maps that serve as the

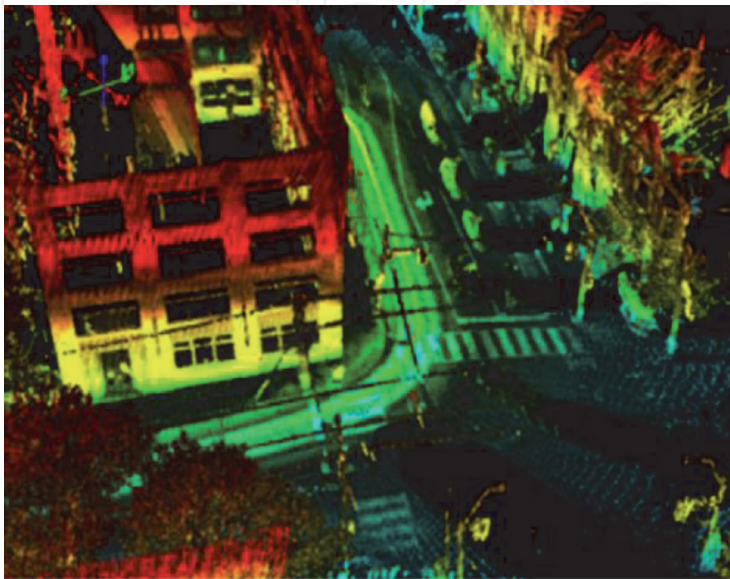


Figure 15.
3D point cloud of part of Ryerson University campus, showing street furniture (color-coded by height).

infrastructure needed for smart cities and smart transportation applications. Besides, Lidar sensors provide AV with its ability to perceive the 3D environment.

In addition, laser scanners can measure the amount of energy that has been reflected from the target after being illuminated with the scanner emitted pulse. The amount of reflected energy from each measured point constitutes the intensity as measured by the sensor. This measurement can prove valuable in a number of applications as asset management and lane marking extraction. The measured intensity depends on the target reflectivity, which affects the amount of energy reflected that can be detected by the sensor. As the target reflectivity decreases, the amount of reflected energy diminishes, thus weakening the signal returned to the sensor and deeming the target undetectable. The range and incidence angle to the target also affect the measured intensity.

Pavement markings have a different reflectance than the asphalt, thus allowing the automated extraction of the markings from the Lidar measured object intensities. This enables lane mapping and also serves as a tool that can be deployed for pavement marking assessment and evaluation. Street signs and furniture can be automatically extracted from the Lidar measured 3D point cloud. This is done using the fusion of the measured intensities, coupled with some machine learning algorithms that uses features geometry which is also measured by Lidar sensors.

A sample of a 3D point cloud intensity showing different features is depicted in **Figure 16**. The numbers in the figure refer to objects as follows: lane markings (1), vehicles (2), road surface arrows (3), zebra crossings (4), bike lane (5), a building (6), and pedestrian (7). Some highlighted features from the Lidar 3D point cloud intensity depicted in **Figure 16a**. As noted, the Lidar data intensity can be very useful in the automated feature extraction of street furniture, pavement marking assessment and evaluation which acts as an enabler for automated asset management. Note that a true color 3D Lidar point cloud can be produced by the fusion of the Lidar 3D point cloud and optical imagery, as shown in **Figure 16b**.

The fusion of Lidar 3D point cloud and optical imagery can prove very useful in such applications as digital twin and 3D visualization with and enhanced visual appeal. In addition, the different modality between Lidar data and optical imagery improves automated 3D point cloud classification as the classification process uses object geometry coupled with reflectance. The continued advances in Lidar sensor technologies, IoT, AI, and big-data algorithms and with faster wireless communications technologies that support cellular data networks 5G (even the 6G networks expected in 2030s), with data transfer rate of speeds of

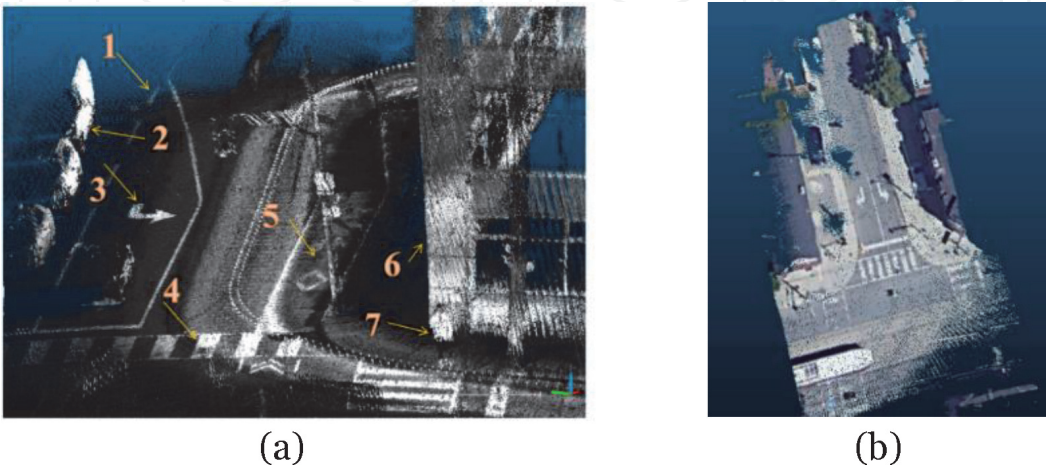


Figure 16.
Lidar 3D point cloud: (a) intensity and (b) true color.

~95 Gb/s, the dream of mass deployment of an advanced driver assistance system level 5 can become a reality.

4. Visibility methodologies

4.1 Infrastructure visibility

4.1.1 High-precision 3D maps

High-precision maps have been identified as one of the key technologies for autonomous driving [26]. These maps need to be purpose-built and highly accurate, with great level of details, and be updated in real-time. These maps, which are particularly made for autonomous driving, provide true-ground-absolute accuracy, normally at centimeter-level (5 cm or better), and contain details organized in multiple map layers, such as base map, geometric map, semantic map, map priors and real-time knowledge [27]. Technologies that can be leveraged for creating such maps include aerial imagery, aerial Lidar data, mobile Lidar mapping, and Unmanned Aerial Vehicles equipped with lightweight sensors. On the ground, HP maps are mainly created using vehicles equipped with high-tech instruments [26]. Efforts on HP mapping in 3D include Civil Maps, HERE, DeepMap, TomTom HP Maps, Mobileye, Uber Localization and Mapping, and Mapper, while most of them follow crowdsourcing models.

In addition to supporting localization and navigation, HP 3D maps can be leveraged for lane network construction, dynamic object detection, prediction of motion of vulnerable road users, and visibility analysis for the correlation between environmental visibility and navigational comfort under autonomous driving [28, 29]. The so-called self-healing mapping systems allows a detailed inventory of road features and objects, part of road infrastructure, on the side of the road for real-time determination of infrastructure visibility.

To provide dynamic routing for any situation, Geographic Information System (GIS) requires updated road network data, real-time traffic information, vehicle current location, and destination. Real-time vehicle location is obtained using Radar and Camera sensors using localization techniques such as simultaneous localization and mapping (SLAM) that can localize vehicle with high precision and map its exact location with respect to the surrounding environment. Localization can not only determine vehicle location, but also map landmarks to update HP maps. Depending on the use of sensors, the algorithm can be Lidar-based, Lidar and camera integrated, or camera-based.

Building and maintaining detailed HP maps in advance presents a very appealing solution for autonomous driving. However, exiting technologies may at the best provide near real-time mapping, and may have missed critical information, such as road markings and dynamic changes of road infrastructure. Real-time SLAM provides better solution to dynamic updating of HP maps for locating vehicles and mapping the surrounding environment.

4.1.2 Sight obstacle detection

The technique for estimating visibility of target points described in Section 2.2 can be extended to detect sight obstacles that restrict infrastructure visibility (e.g. traffic signs). Specifically, as shown in **Figure 7a**, the visibility of the target points are determined by comparing their distances to the sight point with the minimum distance (i.e. d_{min}) of the closest obstacle point to the sight point in each pillar.

The sight obstacles can be detected in a similar way. The respective depth maps of the obstacle points and target infrastructure points are first generated following the steps presented in Section 2.2. Then, at the stage of comparing the depth maps, another parameter is calculated, as shown in **Figure 7a**. Let d_{max} be the maximum distance of the target points from the sight point. The obstacle points farther than d_{max} will not affect the visibility of the target points. On the contrary, the obstacle points whose d -values lie between d_{min} and d_{max} will obstruct the target infrastructure points whose d -values exceed d_{min} . The sight obstacle detection is completed when all pillars are estimated. **Figure 7b** shows the detected sight obstacles that affect the visibility of a traffic sign. In this case, when combined with the techniques for automatically identifying traffic signs from MLS data, the visibility estimation procedure can help understand the visibility of traffic signs to the AV along a road corridor.

4.2 Traffic visibility

4.2.1 Overview

Guaranteeing adequate traffic visibility is crucial to the safe operation of AV. If the sight lines to the conflicting objects (e.g. vehicles, pedestrians, and cyclists) are obstructed by some obstacles (**Figure 17a**), the AV may not identify the objects at a safe distance, which may lead to a collision. It is quite challenging for AV vehicles to predict where pedestrians or cyclists may be present in a complex road scene in real-time [30]. In this case, it is meaningful to identify the positions where road users may exist in real-world using dense MLS points and investigate the visibility of these locations to the AV in advance (see **Figure 17a** and **b**). Then the georeferenced visibility information can be incorporated into HP maps, which may help AV make some proactive measures to reduce the collision risk at locations with unsatisfactory traffic visibility.

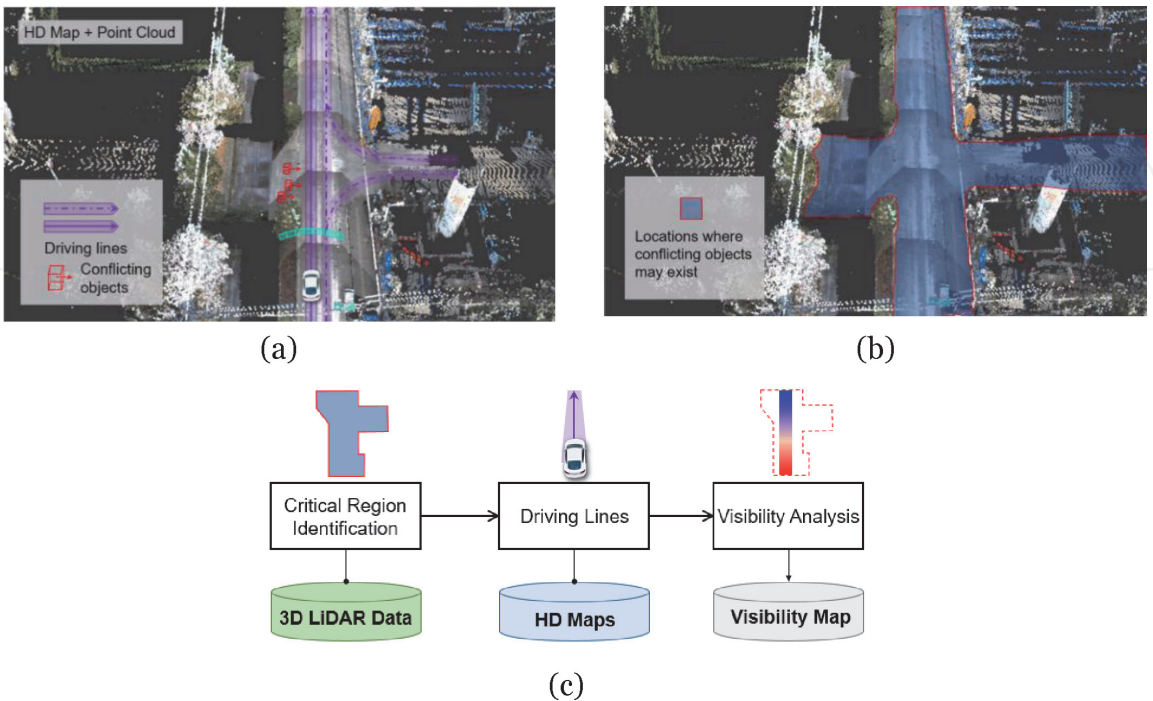


Figure 17. Traffic visibility estimation for AV: (a) reasons for estimating traffic visibility, (b) critical positions, and (c) general workflow of traffic visibility estimation.

As shown in **Figure 17c**, the general process of traffic visibility analysis comprises two main components: (1) critical positions and (2) driving lines. The critical positions correspond to the target points in the VE process (see **Figure 4**). The driving lines in HP maps which aid in navigating AV can help derive sight points in this case. Traffic visibility estimation in autonomous driving involves estimating the visibility of some critical positions along the pre-defined driving lines to the AV and generating the visibility map.

4.2.2 Critical position identification

In this example, the planar ground points are identified as the locations where vehicles, pedestrians, or cyclists may exist. A common workflow shown in **Figure 8a** is used to extract the critical positions. More specifically, MLS data are first partitioned into a number of pillars with the linear-index based segmentation technique illustrated in **Figure 11**. Next, a pillar-wise filtering is performed that the points with δ_h (user-defined, e.g., 0.2 m) higher than the lowest point are removed. The remaining points are considered as the rough ground points. Then, the kd-tree data structure is applied to find k-nearest neighbors for each data point. The neighboring points are used to derive the normal vector $\vec{\xi}$ of each point [31]. Let γ be the angle between $\vec{\xi}$ and the vertical direction (i.e. $(0, 0, 1)^T$). The points with $\gamma \leq 5^\circ$ are considered as the horizontal and planar points.

Because we mainly focus on the locations where the road users may exist and may conflict with the AV, a distance-based segmentation method is applied to remove some isolated planar point clusters. The point cluster with the largest size is determined as the ‘critical positions’. MLS data of a 500 m long urban road section is used to illustrate the process of VE. The extracted critical positions using the procedure shown in **Figure 8a** are marked in red in **Figure 8b**. The critical positions covers pavement surface, sidewalks, and some planar surfaces connected to the pavement. It is also recommended to use the semantic segmentation techniques powered by deep learning to identify the critical positions more accurately.

4.2.3 Visibility estimation

The driving lines of AV are also plotted in **Figure 8b**. Each driving line is composed of many consecutive nodes. At each node, suppose that the sight point overlap with the node on the horizontal plane. However, the elevation of sight point h_s is adjustable to accommodate different height of Lidar sensors mounted on the autonomous driving vehicles. Then, the VE procedure described previously is applied to estimate the visibility of critical positions to the sight point. **Figure 9** presents the VE results at a single node. The visible and invisible critical positions are marked in green and red, respectively, while the potential obstacle points are marked in gray. In **Figure 9**, the range of φ is $[-180^\circ, 180^\circ]$, where both the front and rear positions are estimated. Different limits can be set on the range of φ to simulate varied horizontal viewing angles. The VE procedure at a single node can be extended to estimate the locations where the sight triangle is clear at intersections [16]. Specifically, if there are invisible (red) regions (see **Figure 9**) inside the sight triangle, it means the sight triangle is not clear.

The VE procedure is executed along the driving lines node by node (see **Figure 10a**) to gain a better understanding of traffic visibility for AV. In this phase, the variables involved are $h_s = 1.6$ m, $d_{view} = 100$ m, $\varphi_{view} = -60$ to 60° , $\phi_{view} = -30$ to 30° , $\delta\varphi = 0.1^\circ$, and $\delta\phi = 0.1^\circ$. The users may also adjust d_{view} , φ_{view} etc. to investigate different situations.

The VE results are visualized in **Figure 10**. In this example, two types of quantitative information are derived based on VE results at each node. **Figure 10b** maps the cumulative times of invisibility N_{invis} at a point level. The initial N_{invis} of each target point is zero. During the process of VE, $N_{invis} = N_{invis} + 1$ each time when the target point is not seen by the ego vehicle. A large N_{invis} means the target point is invisible to the AV for many times. The magnitude of N_{invis} is visualized with colors in **Figure 17b** which may help to identify the possible locations of the blind areas for the ego vehicles. As marked with rectangles, the visibility of these locations are poorer, which means that the AV may need to decelerate when approaching these locations to avoid potential right-angle collisions. Also, the blind-area results can be estimated in conjunction with collision data in future studies to better understand road safety.

Figure 10c shows the variation of the visibility ratio V_r along the driving lines. The V_r measures the ratio of visible targets to all targets at a single position. A low V_r may indicate that a majority of target points are invisible to the AV. Because V_r can be integrated with the driving lines, the AV can know where the visibility ratio is relatively low based on its location. In that case, the ego vehicle can decelerate in advance at a location with very low V_r to reduce the probability of colliding with a running pedestrian from a blind area.

5. Decision

5.1 Path planning

This chapter has addressed so far the visibility element of autonomous vehicles, which is the main focus of the chapter. It is useful, however, to highlight the decision element (see **Figure 1**), which involves path planning. This element is considered to be the main challenge of autonomous vehicles. Path planning enables the autonomous vehicle to find the safest, most convenient, and most economic route between two points. One of the main modeling approaches to define such a route is Model Predictive Control (MPC). There is a variety of MPC, but basically the model solves a finite-time constrained optimal control problem in a receding horizon using nonlinear optimization. An application of path planning and control for overtaking on two-lane highways using nonlinear MPC can be found in [32].

There are two approaches for path planning (hierarchical and parallel) [33]. In the hierarchical approach the autonomous vehicle completes long-term mission and reduce the workload of motion planning. The input higher-level mission is decomposed into sub-missions and then passed to the next level down. The hierarchical model helps to resolve many complicated problems, yet it might slow down the work of a vehicle's feedback control and complicate the performance of sophisticated maneuvers. In the parallel approach, the tasks are more independent and can proceed simultaneously. In this approach, each controller has dedicated sensors and actuation mechanisms. The advantages of this approach: (1) the controllers are running at high frequency, which makes them safe and stable, (2) a high level of smoothness and performance is achieved by the controllers, and (3) the approach is relatively inexpensive and does not require using complicated motion planning devices. However, for some purposes, the hierarchical approach is more efficient. **Figure 18** illustrates the hierarchical approach, based on [33].

Extensive research has been conducted for path planning for automated parking, which is now a reality in many cities [34–38]. Automated parking aims to eliminate the influence of human factors, improve the quality and accuracy of control, and reduce the maneuver time by optimizing vehicle path in restricted parking zones. The vehicle to anything technology (V2X) allows communication between AV and

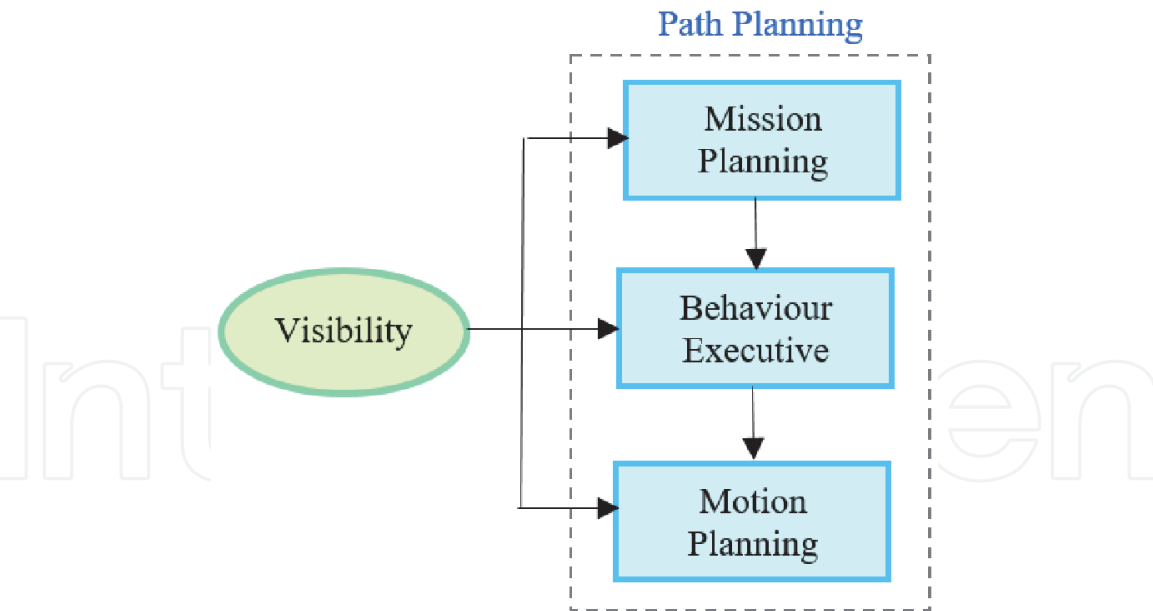


Figure 18.
Hierarchical approach for path planning.

a building through data exchange to find an unoccupied space and generate the route to a destination. For large-sized trucks, the tasks involve predicting stable and safe passing on road curves and forecasting precise control for docking. Numerous approaches that consider different control strategies, sensory means, and prediction algorithms (e.g. geometric, fuzzy, and neural) for predicting vehicle parking path have been developed.

Recently, the bicycle kinematic models for vehicle motion have been used for path planning of automated parking [39]. Three basic types of vehicles were considered: passenger car, long wheelbase truck, and articulated vehicles with and without steered semitrailer axes. The authors presented a system of differential equations in matrix form and expressions for linearizing the nonlinear motion equations that increased the speed of finding the optimal solution. An original algorithm that considered numerous constraints was developed for determining vehicle permissible positions within the closed boundaries of the parking area using nonlinear MPC that finds the best trajectories.

Figure 19 shows the kinematics of the curvilinear motion and the simulation results that validated the proposed model. Note that in this study, kinematic vehicle models were used instead of dynamic models. Kinematic models assume that no slip

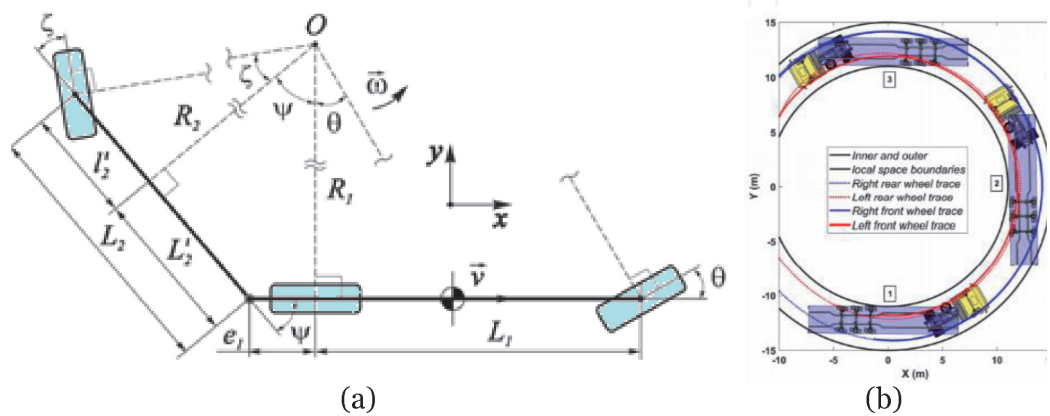


Figure 19.
Modeling and simulation of autonomous docking of tractor-semitrailer vehicle with semitrailer's steered axes: (a) kinematics of curvilinear motion and (b) simulation results [39].

occurs between the wheels and the road. This assumption is reasonable for vehicles moving at low speeds, which is the case for parking. However, dynamic models should be used for path planning of autonomous vehicles on highways since they are more accurate.

5.2 Intelligent car-following

Autonomous vehicles and connected automated vehicles (CAV) with advanced embedded technology can deliver safe and effective traffic movements [40]. However, there will be a transition period when AV and human-driven vehicles would share the common road space. Therefore, it would be extremely critical to organize the interactions which will likely be generated from that vehicle mix. Due to the anticipated asymmetric response of these two modes of vehicles, many possible combinations of these vehicles are possible. These combinations may give rise to interactions due to longitudinal and transverse movements.

To ensure safe interactions, the intelligent vehicles may include adaptive cruise control (ACC) and cooperative adaptive cruise control (CACC) systems. These systems mainly assist the acceleration control for the longitudinal movements. The systems control the acceleration based on the distance gap and the speed difference between the current vehicle and the vehicle ahead (leader), where the vehicle accelerates and decelerates based on the speed changes of the leader. For the ACC system, the distance and speed are obtained using radar, Lidar, or video cameras. For the CACC system, V2V communications are used to share the acceleration, deceleration, braking capability, and vehicle positions [41]. This communication significantly shortens the time headway (0.5 s) of the CACC vehicle compared to the ACC vehicle (1.4 s).

To obtain deeper insights into car-following behavior, micro simulation studies were conducted to estimate the impacts of AV and CAV using a variety of assumptions [41]. Car-following models involving intelligent vehicles were developed to evaluate their traffic impacts on such aspects as capacity and level of service, traffic stability, travel time, and vehicle speed. Several studies estimated the energy, environmental, and safety impacts using surrogate safety measures, such as travel speed, time-to-collision, and post-encroachment-time.

New car-following models for AV and CAV were developed by modifying available traditional car-following models to mimic the intelligent vehicle characteristics. The Intelligent Driver Model (IDM) [42], a time-continuous car-following model for the simulation of freeway and urban traffic, is the most commonly used model for simulations of intelligent vehicles. The basic function of the model is given by

$$a_{IDM}(s, v, \Delta v) = \frac{dv}{dt} = a \left[1 - \left(\frac{v}{v_o} \right)^\delta - \left(\frac{s^*(v, \Delta v)}{s} \right)^2 \right] \quad (9)$$

$$s^*(v, \Delta v) = s_o + vT + \frac{v\Delta v}{2\sqrt{ab}} \quad (10)$$

where s = current distance gap to the leader (m), s_o = minimum distance gap (m), v = current speed (m/s), v_o = desired safety speed (m/s), Δv = speed difference between the current vehicle and the leader (m/s), δ = parameter that determines the magnitude of acceleration decrease (usually set equal to 4), T = desired safety gap (s), a = maximum acceleration (m/s^2), and b = comfortable deceleration rate (m/s^2). Note that $s^*(v, \Delta v)$ is the desired distance gap (m).

Another car-following model that is commonly used models for conducting cooperative intelligent-vehicle simulations is the MICroscopic Model for Simulation

of Intelligent Cruise Control (MIXIC) [43]. Both IDM and MIXIC have been used as the benchmark models for combined AV and human-driven vehicles for various market penetration rates ranging from 0 to 25% [44, 45]. However, detailed calibration procedures of model parameters used in these models are not available due to the limited or unavailability of empirical data.

On these lines, substantiating vehicle's longitudinal motion in mixed traffic conditions is even more critical using available car-following model parameters. In the present context, the car-following models used in simulation packages need to be suitably modified. Further, most of these models are straightforward in their approach with a limited set of parameters. The simplification of available model parameters might not affect the simulation results to the extent the intelligent vehicles may impact the traffic streams in real-field experiments. For example, most car-following models tend to have fixed parameters and will average out driving characteristics. If such models are used for AV, they can reflect neither the driving style of the vehicles' actual drivers nor the contexts in which they drive.

In this direction, the intelligent car-following behavior models, which can concurrently consider all necessary parameters and adapt to specific field actions must successfully integrate possible mixed traffic scenarios. Hence, there is a strong need to modify the existing conventional models to handle the higher order of stochasticity due to mixed driving conditions. Alternatively, data-driven methodologies can be used to provide high-quality trajectory data for varying scenarios of automation. These data may be generated from experiments by creating test beds or even using driving simulators with human-machine interaction. For this purpose, the trajectory data available from simulation models may also be useful. Considering the recent highlights of AI, even intelligent car-following models can benefit from data learning methodologies. Clearly, the impact analysis of intelligent vehicles should consider the many uncertainties due to mixed traffic conditions. Although a few new models have been developed or modified to incorporate car-following and lane-changing attributes of intelligent vehicles, empirical data are warranted for model calibration.

6. Conclusions

This chapter has presented an overview of the visibility-based technologies and methodologies for autonomous driving. Based on this overview, the following comments are offered:

1. The Lidar height is a key parameter that affects the visibility of the road ahead and in turn sight distance that influences the design of horizontal and vertical alignment as well as intersections. To ensure safety of AV when operating with human-driven vehicles on existing highways, the Lidar height should generally be larger than the current design heights for passenger cars and less than that for trucks.
2. The new Lidar system developed at Ryerson University holds great potential for generating 3D maps for AV. The system is cheap, allows AV a robust 3D perception of the environment, and allows obstacle detection in a fast and reliable manner even at long ranges.
3. The infrastructure and traffic visibilities can be estimated for autonomous vehicles based on a combination of MLS data and the driving lines of HP maps. The generated visibility map can be incorporated into the HP maps and help

the autonomous vehicle develop some proactive speed control strategies at locations where the visibility is unsatisfactory.

4. The uncertainty of the input variables can substantially affect the reliability of the design elements. Therefore, reliability analysis should be incorporated in all the tasks related to SD determination to ensure safe operation of the autonomous vehicles. Extensive uncertainty research has been conducted on the required SD and similar research is lacking for the available SD determined using the Lidar.
5. Geographic information systems provide support for route planning and real-time and dynamic routing/navigation of AV using GNSS, localization techniques, and HP maps. The GIS navigation helps to guide the autonomous vehicle along the best route to its destination safely.
6. The intelligent car-following strategies are expected to show good performance in simulation environments. However, in the calibration using real-field conditions/experiments, the input to simulation models would not be beneficial as it is limited to the available parameters that correspond to specific conditions. The parameters should be based on the surrounding environmental factors for all possible combinations of autonomous and human-driven vehicles. This is a mammoth challenge due to the limited empirical data.
7. Another challenge of autonomous driving is related to the complexity of the software algorithms needed to process the large amount of information coming into the AV (estimated at 1 billion lines of code) which is then used to make decisions about the proper action. The software complexity is compounded by the fact that AV will operate on roads full of unpredictable drivers of human-driven vehicles.

Acknowledgements

This chapter is financially supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

List of acronyms

ACC	adaptive cruise control
AI	artificial intelligence
ALS	airborne laser scanning
AV	autonomous vehicles
CACC	cooperative adaptive cruise control
CAV	connected automated vehicles
DSD	decision sight distance
GIS	geographical information system
GNSS	global navigation satellite system
HP	high-precision
IDM	intelligent driver model
IoT	internet of things
ISD	intersection sight distance
Lidar	light detection and ranging

LOS	line of sight
MLS	mobile laser scanning
MSS	mobile mapping system
PSD	passing sight distance
SD	sight distance
SLAM	simultaneous localization and mapping
SSD	stopping sight distance
RSS	remote sensing sensor
VE	visibility estimation
V2I	vehicle-to-infrastructure
V2V	vehicle-to-vehicle
1D	one dimensional
3D	three dimensional

Author details

Said Easa^{1*}, Yang Ma², Ashraf Elshorbagy¹, Ahmed Shaker¹, Songnian Li¹ and Shriniwas Arkatkar³

1 Ryerson University, Toronto, Canada

2 Southeast University, Nanjing, China

3 Sardar Vallabhbhai National Institute of Technology, Surat, Gujarat, India

*Address all correspondence to: seasa@ryerson.ca

IntechOpen

© 2020 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Shladover S. Connected & automated vehicle systems: Review. *Intelligent Transp. Sys. J*, 22(3), 2018.
- [2] Allied Market Research. *Autonomous Vehicle Market by Level of Automation, Component, and Application*. 2019. <https://www.alliedmarketresearch.com/autonomous-vehicle-market>.
- [3] NCHRP *Connected & autonomous vehicles and transportation infrastructure readiness*. Project 20–24(111), TRB, Washington, DC, 2017.
- [4] Mohamed A. Literature survey for autonomous vehicles: Sensor fusion, computer vision, system identification, and fault tolerance. *Inter. J. Auto. Control*, 12(4), 2018.
- [5] Al-Qaysi Q, Easa S M, Ali N. Proposed Canadian automated highway system architecture: object-oriented approach. *Can. J. Civ. Eng.*, 30(6), 2003.
- [6] Shladover S. Automated vehicles for highway operations (automated highway systems). *IME Proceedings, Part I: Journal of Systems and Control Engineering*, 219(1), 53-75, 2005.
- [7] Easa S M. *Automated highways*. In Encyclopedia of Electrical and Electronics Engineering, John Wiley & Sons, New York, N.Y., 2007.
- [8] American Association of State Highway and Transportation Officials. *A policy on geometric design of highway and streets*. AASHTO, Washington DC, 2008.
- [9] Easa S M. *Geometric design*. In Civil Engineering Handbook, W.F. Chen and J.Y. Liew eds., CRC Press, Boca Raton, FL, Chapter 63, 2002, 1-39.
- [10] Transportation Association of Canada. *Geometric design guide for Canadian roads*. TAC, Ottawa, Ontario, 2017.
- [11] Khoury J et al. Initial investigation of the effects of AV on geometric design. *Journal of Advanced Transportation*, Vol. 2019, 2019, 1–10
- [12] Ma Y, Zheng Y, Cheng J, Zhang Y, and Han W. A convolutional neural network method to improve efficiency and visualization in modeling driver's visual field on roads using MLS data. *Transportation Research Part C: Emerging Technologies*, 106, 2019, 317–344.
- [13] Ma Y, Zheng Y, Cheng J, and Easa S M. Analysis of dynamic available passing sight distance near right-turn horizontal curves during overtaking using Lidar Data. *Can J Civ. Eng.*, 2019.
- [14] Ma Y, Zheng Y, Cheng J, and Easa S M. Real-time visualization method for estimating 3D highway sight distance using Lidar data. *J. Transp. Eng. - Part A: Systems*, 2019.
- [15] Gargoum S A, El-Basyouny K, and Sabbagh J. Assessing Stopping and Passing Sight Distance on Highways Using Mobile Lidar Data. *J. Compt. Civ. Eng.*, 32(4), 04018025, 2018.
- [16] Jung J, Olsen M J, Hurwitz D S, Kashani A G, and Buker K. 3D virtual intersection sight distance analysis using Lidar data. *Transportation Research Part C: Emerging Technologies*, 86(JAN.), 563–579, 2018.
- [17] Shalkamy A, El-Basyouny K, and Xu H Y. Voxel-Based Methodology for Automated 3D Sight Distance Assessment on Highways using Mobile Light Detection and Ranging Data. *Transportation Research Record*, 2674(5), 587–599, 2020. doi:10.1177/0361198120917376.
- [18] Zhang S X, Wang C, Lin L L, Wen C L, Yang C H, Zhang Z M, and Li J. Automated visual recognizability evaluation of traffic sign based on 3D

Lidar point clouds. *Remote Sensing*, 11 (12), 2019. doi:10.3390/rs11121453.

[19] MATLAB. 2020. sub2ind: Convert subscripts to linear indices. Retrieved from <https://www.mathworks.cn/help/matlab/ref/sub2ind.html?lang=en> (accessed on November 1, 2020).

[20] OpenDSA. 2020. CS3 Data Structures and Algorithms. 2020, <https://opensa-server.cs.vt.edu/ODSA/Books/CS3/html/LinearIndexing.html> (accessed on November 1, 2020).

[21] Ma Y, Zheng Y, Easa S M, and Cheng J. Semi-automated framework for generating cycling lane centerlines on roads with roadside barriers from noisy MLS data. *ISPRS J. Photo. Remote Sensing*, 167, 2020, 396–417.

[22] Shan J, and Toth C K. *Topographic laser ranging and scanning: Principles and processing*. CRC Press, Boca Raton, FL, 2018.

[23] Elshorbagy A. *A Crosscutting Three-modes-of-operation Unique LiDAR-based 3D Mapping System, Generic Framework Architecture, Uncertainty Predictive Model and SfM Augmentation*. Doctoral Dissertation, Department of Civil Engineering, Ryerson University, Toronto, Canada, 2020.

[24] Shaker A and Elshorbagy A. Systems and methods for multi-sensor mapping. Application Number 62/889,845–24440-P58819US00, 2019.

[25] Shaker A and Elshorbagy A. Systems and methods for multi-sensor mapping using a single device that can operate in multiple modes. PCT Patent Application No. PCT/CA2020/051133, 2019.

[26] Seif H G and Hu X. Autonomous driving in the iCity—HD maps as a key challenge of the automotive industry. *Engineering*, 2(2), 2016, 159–162.

[27] Vardhan H. HD Maps: New age maps powering autonomous vehicles.

Geospatial World, 2017 <https://www.geospatialworld.net/article/hd-maps-a-utonomous-vehicles> (accessed on November 1, 2020).

[28] Chou F C, Lin T H, Cui H, Radosavljevic V, Nguyen T, Huang T K, and Djuric N. Predicting motion of vulnerable road users using high-definition maps and efficient convnets,. 2019, arXiv preprint arXiv:1906.08469.

[29] Morales Y, Even J, Kallakuri N, Ikeda T, Shinozawa K, Kondo T, and Hagita N. Visibility analysis for autonomous vehicle comfortable navigation. *IEEE International Conference on Robotics and Automation*, 2014, 2197–2202.

[30] Ahmed S. Pedestrian/cyclist detection and intent estimation for AV: A survey. *App. Sc.*, 9, 2019.

[31] Yang B S, Liu Y, Dong Z, Liang F X, Li B J, and Peng X Y. 3D local feature BKD to extract road information from mobile laser scanning point clouds. *Journal of Photogrammetry and Remote Sensing*, 130, 2017, 329–343, doi: 10.1016/j.isprsjprs.2017.06.007.

[32] Easa S M and Diachuk M. Optimal speed plan for overtaking of a autonomous vehicles on two-lane highways. *J. Infrastructures*, 5(44), 2020, 1–25.

[33] Ryabchuk P. How Does Path Planning for Autonomous Vehicles Work? 2020, <https://dzone.com/users/3246906/paulryabchuk.html> (accessed on November 1, 2020).

[34] Lee H, Chun J, and Jeon K. Autonomous back-in parking based on occupancy grid map and EKF SLAM with W-band radar. *Proc., International Conference on Radar*, Brisbane, Australia, 2018, 1–4.

[35] Lin L and Zhu J J. Path planning for autonomous car parking. In *Proceedings of the ASME Dynamic Systems and*

Control Conference, Vol. 3, Atlanta, GA, USA, 2018.

Intelligent Transportation Systems, 7(4), 2006, 429–436.

[36] Kiss D and Tevesz G. Autonomous path planning for road vehicles in narrow environments: An efficient continuous curvature approach. *J. Adv. Transp.*, 2017.

[44] Kesting A, Treiber M, Schönhof M, and Helbing D. Adaptive cruise control design for active congestion avoidance. *Transportation Research Part C: Emerging Technologies*, 16(6), 2008, 668–683.

[37] Wang Y, Jha D K, and Akemi Y. A two-stage RRT path planner for automated parking. *Proc., 13th IEEE Conference on Automation Science and Engineering*, Xi'an China, 2017, 496–502.

[45] Talebpour A and Mahmassani H S. Influence of connected and autonomous vehicles on traffic flow stability and throughput. *Transportation Research Part C: Emerging Technologies*, 71, 2016, 143–163.

[38] Ballinas E, Montiel O, Castillo O, Rubio Y, and Aguilar L T. Automatic parallel parking algorithm for a car-like robot using fuzzy PD+1. *Control. Eng. Lett.*, 26, 2018, 447–454.

[39] Diachuk M, Easa S M, and Bannis J. Path and control planning for a autonomous vehicles in restricted space and low speed. *J. Infrastructures*, 5(4), 2020, 1–27.

[40] Greer H, Fraser L, Hicks D, Mercer M, and Thompson K. *Intelligent transportation systems benefits, costs, and lessons learned*. U.S. Dept. of Transportation, ITS Joint Program Office, 2018.

[41] Wooseok D, Omid M R, Luis M-M. Simulation-based connected and automated vehicle models on highway sections: A Literature review. *Journal of Advanced Transportation*, 2019. <https://doi.org/10.1155/2019/9343705>.

[42] Treiber M, Hennecke A, and Helbing D. Congested traffic states in empirical observations and microscopic simulations. *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics*, 62(2), 1805–1824, 2000.

[43] Van Arem B, Van Driel C J, and Visser R. The impact of cooperative adaptive cruise control on traffic-flow characteristics. *IEEE Transactions on*