Review

Collision Risk in Autonomous Vehicles: Classification, Challenges, and Open Research Areas

Pejman Goudarzi [1,*] and Bardia Hassanzadeh [2]

[1] Information Technology Faculty, ICT Research Institute (ITRC), Tehran 14155-3961, Iran
[2] Faculty of engineering, University of Alberta, Edmonton, AB T6G 2R3, Canada; bardia@ualberta.ca
* Correspondence: pgoudarzi@itrc.ac.ir

Abstract: When car following is controlled by human drivers (i.e., by their behavior), the traffic system does not meet stability conditions. In order to ensure the safety and reliability of self-driving vehicles, an additional hazard warning system should be incorporated into the adaptive control system in order to prevent any possible unavoidable collisions. The time to contact is a reasonable indicator of potential collisions. This research examines systems and solutions developed in this field to determine collision times and uses various alarms in self-driving cars that prevent collisions with obstacles. In the proposed analysis, we have tried to classify the various techniques and methods, including image processing, machine learning, deep learning, sensors, and so on, based on the solutions we have investigated. Challenges, future research directions, and open problems in this important field are also highlighted in the paper.

Keywords: collision risk; time-to-collision; autonomous vehicles; machine learning

check for updates

Citation: Goudarzi, P.; Hassanzadeh, B. Collision Risk in Autonomous Vehicles: Classification, Challenges, and Open Research Areas. *Vehicles* **2024**, *6*, 157–190. https://doi.org/10.3390/vehicles6010007

Academic Editor: Xianke Lin

Received: 18 November 2023
Revised: 4 January 2024
Accepted: 10 January 2024
Published: 12 January 2024

1. Introduction

Autonomous vehicles and driver-assistance systems are becoming a reality with the rapid development of wireless communication and sensors. As such, many interesting questions arise. For example, how can sensors be used to control a car? How much can autonomous vehicles improve traffic conditions, such as reducing collisions, stabilizing traffic flow, and increasing highway throughput? As autonomous driving systems can take into account more environmental information than human drivers, they offer attractive improvements over today’s traffic situation. At least the early versions of self-driving cars still appear to many as if they were designed for human drivers.

Even with the same control mode, there is still a significant difference between self-driving vehicles and conventional cars. Drivers are unable to respond to hazards as quickly as sensors do. The safe space that needs to be maintained between self-driving cars (proportional to their reaction time) is much smaller than that between conventional vehicles. In addition, self-driving cars will have a much higher density of traffic than today’s vehicles.

A decrease in reaction time makes controlling the car more difficult, as does maintaining stability. For autonomous driving, collision detection and avoidance are important parts of the perception and planning system. The future estimation of crash risk is mainly for automated driving systems, as well as for driver-assistance systems. It is essential that the vehicles’ systems understand, identify, and track their surrounding objects and are aware of collisions. Drivers use some vehicle-related data such as relative velocity information to calculate their own acceleration relative to the leading car. Obtaining reliable and stable measurements of relative velocity is therefore key to self-driving cars.

Other than the distance of vehicles, the collision risk depends also upon a number of different sensors and technologies that are employed [1–3]. For example, technologies such as Bluetooth low energy (BLE), ZigBee, ultra-wide bandwidth radio (UWB), dedicated short-range communication (DSRC), cellular vehicle to everything (C-V2X), and 5G-NR V2X have been employed in autonomous vehicles (AVs) and may affect the risk of collision [1]. On the other hand, different driving strategies such as defensive, negotiated, competitive, and cooperative strategies may have effects on the risk of collision. Furthermore, the type of sensors and software used in AVs such as ultrasonic sensors, radio detection and ranging (RADAR), mmWave with THz, light detection and ranging (LiDAR), global navigation satellite systems (GNSS), high-definition (HD) maps, ultrasounds, and cameras have their own role in the collision risk of the vehicles [3].

The general architecture of autonomous connected vehicles (ACV) has been investigated in [2]. Specifically, it is discussed how new and trending technologies such as multi-access edge computing (MEC), 5G new radio (5GNR), software-defined networks/network function virtualization (SDN/NFV), blockchain, network slicing, and federated learning influence the collision risk of autonomous vehicles [2].

In summary, the total number of papers that are investigated in this survey is 68, and the percentage covering each publisher is depicted in Figure 1.

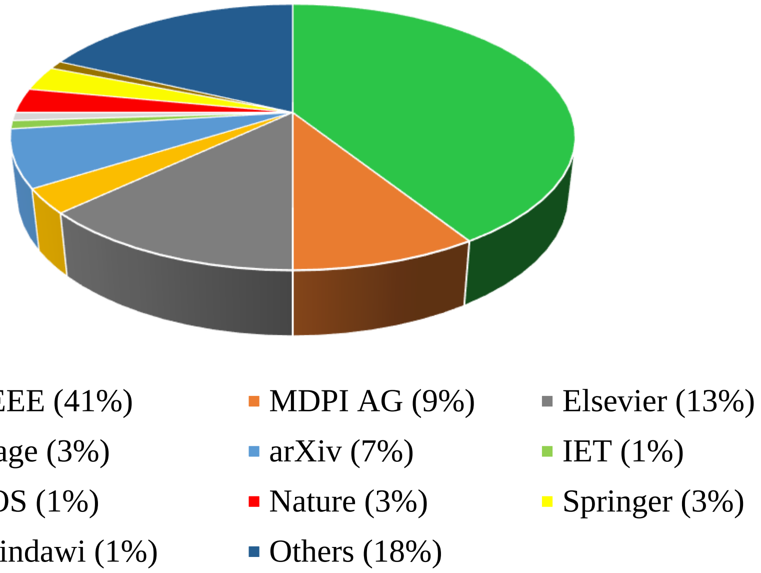

Figure 1. Percentage of manuscripts coverage between different publishers.

The rest of the paper is organized as follows. In Section 2, we review and categorize the methods and techniques used in the time-to-collision estimations of vehicles. In Section 3, we review and categorize the methods and techniques used in the collision risk estimations of vehicles. Section 4 is about data and methods. Sections 5 and 6 are dedicated to open research areas and challenges in the field. Finally, in Section 7, some concluding remarks are presented.

2. Time to Collision

By definition, the time to collision (TTC) indicates how long an observer takes to make contact with a surface under constant velocity. The research work in this context can be categorized into three major domains, which are computer vision-based techniques, artificial intelligence (AI)-based techniques (supervised learning, unsupervised learning, reinforcement learning), and other miscellaneous techniques that are mainly based on mathematical formulation and signal processing tools.

2.1. Computer Vision-Based Techniques

A new approach to calculating TTC is presented in [4] using affine scales calculated from active lines. The results of this study were compared with two recently proposed criteria: scale-invariant ridge segment (SIRS) and image brightness derivative (IBD). During an approximate associative transform of the image flow, the active contour affine scale (ACAS) provides a better estimate of TTC, while IBD overestimates the collision time, and SIRS provides a reliable estimate. In some cases, this may not be true. Therefore, ACAS is a more accurate scale. The only disadvantage of the ACAS method is that the obstacles must first be separated from the background. ACAS allows for the tracking of multiple ridges in real time and the flagging of ridges with small TTCs for a more accurate calculation of TTC. The SIRS would thus be able to detect and initialize potential obstacles and then use an affine scale to estimate the obstacle with more accuracy. It is possible to improve the collision time by smoothing over time with a smoothing function such as a Kalman filter or other methods. It has been found that recursive least squares (RLS) with a forgetting parameter produces good results. As a result of such filtering, collision time estimation may be delayed, which may cause navigation problems in real time.

In [5], contact time is estimated through image sequences using a featureless method. The method does not require the extraction and tracking of features, so its calculations are more efficient than those based on features. In a mobile robot platform, an error-based controller with gain timing is coupled with a feature-free estimation algorithm. In order to maintain a reference time for collisions, the speed of the robot moving is controlled successfully. Additionally, angular velocities have been incorporated into the feature-free estimation algorithm. As a result of the Kalman filter, the estimation algorithm and the control strategy perform better.

Furthermore, the central limit theorem shows that the models are more reliable with more data, but as shown in this paper, they also accurately predict behavior with limited data. In the work [6], a new mathematical constraint between contact time, acceleration, and depth, called τ constraint, was presented. Several experiments were conducted in a path estimation task using inexpensive and commercially available sensors to verify the practical effectiveness of the constraint (See Figure 2).

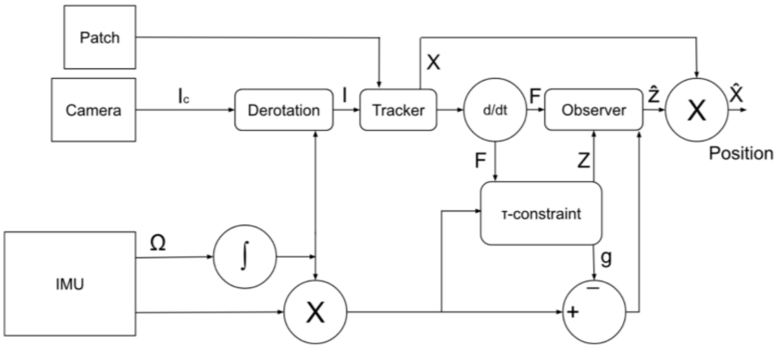

Figure 2. τ constraint declaration [6].

While achieving 30–50% lower absolute path error than two commonly used state-of-the-art visual inertial odometry (VIO) methods, this method ran 6.8–27 times faster. Most of the speed increase is due to the use of only a small part of the image.

In [7], a new technique was presented for parameter fitting of the model equations to improve and stabilize the collision time estimates. Furthermore, several methods have been developed to extend its application to arbitrary motions. With the use of a robot equipped with a webcam in an indoor environment, this approach has been successfully tested.

A machine vision-based approach to relative speed estimation is presented in [8]. The first step in calculating the relative speed is to estimate where the collision time occurs, and then multiply the measured distance by the inverse of the collision time (1/TTC). An Android smartphone is used to run the entire system in real time. A time-varying image does not provide a good estimation of collision time unless significant filtering is used in order to improve the accuracy of the estimation. However, it can be used in conjunction with other measures of relative velocity to provide valuable information.

The vibration of the camera installed in the car may result in blurry images, which negatively affects the estimation of timelapse.

An approach for calculating contact time using photometric information is presented in [9]. In a scene with a moving light source, the observed intensity changes as the light source moves. An analysis of the intensity change in the camera images was conducted in this article, and it was revealed that the contact time could be estimated from the light intensity change. First, a basic method for estimating the contact time of a light source was derived from changes in image intensity. Additionally, this method does not require information such as light source radiation, object reflections, and object surface orientations.

A proposed vehicle detection system based on point detection is presented in [10]. Calculating collision time and distance can be accomplished very precisely by calculating the centers of gravity of nearby spots. Night vision applications are generally affected by rain or fog on the windshield, which causes light sources such as car headlights to blur and smear heavily. An additional disadvantage is the asymmetric shape change caused by the blinking of the leading vehicle.

In [11], the authors used computer vision techniques to estimate the time to collision.

2.2. AI-Based Techniques

A collision time estimation algorithm proposed in [12] was shown to be accurate, hardware compatible, and potentially implementable on smart video sensor hardware. Using biological motion energy features and random forests, the algorithm estimates TTC from dense optical flow. It has been shown that the random forests method is more accurate at estimating optical flow and consumes fewer computational resources than the voting method in previous works. Using this algorithm allows for flexible tradeoffs between estimation accuracy, processing speed, and resource consumption in a scalable pixel pipeline architecture that is hardware compatible.

The information provided by collision time can be more useful in certain scenarios than depth information. Current methods of estimating collision times, however, either make impractical assumptions or cannot be implemented in real time. In [13], a framework for estimating collision time from a monocular camera input is presented. It computes a geofence time in just 6.4 milliseconds to detect objects predicted to collide at a given time. In the process of computing a number of these geofences, it can estimate the collision time with arbitrary quantization, including a continuous collision time.

As proposed in [14], the integration of events and depth data allows for better collision avoidance strategies. In simple static scenes, depth sensors are accurate, but they interfere with fast-moving objects. However, cameras do not work well on texture-free surfaces, since they lack a certain temporal resolution and dynamic range. The strengths of each sensor method are balanced by the weaknesses of the other. This article presents an approach to estimating the impact time for each pixel that overcomes these issues. It was demonstrated that the resulting time-to-impact (TTI) maps can guide a collision control policy better than those based on background segmentation. A schematic view of the proposed obstacle avoidance system in [14] is presented in Figure 3.

As illustrated by human driving ability, this method [15] calculates the collision time based on the motion of a set of linear features, which can be applied to any background or vehicle. By preventing complex video searches, detecting vehicles, and estimating depth, it can provide computational efficiency for real-time processing. Using a single video camera, spatial–temporal region and motion filtering have improved the accuracy of motion estimation. It should be noted that this movement in non-flowing directions causes a quick warning of a possible collision in all directions if it is not handled accordingly. Using only motion, this method has been tested on videos and in different driving environments.

In [16], the authors used machine learning techniques to estimate the time to collision.

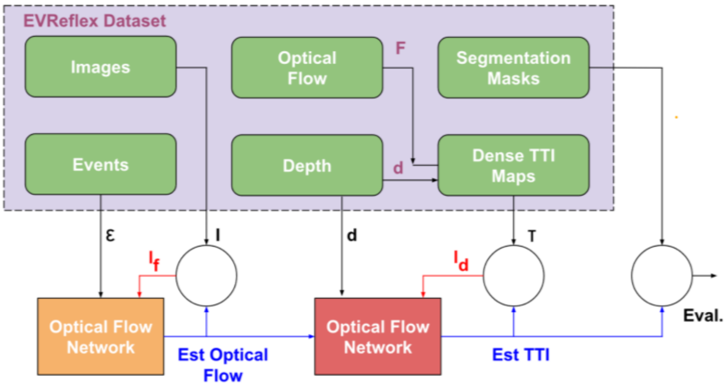

Figure 3. A sample obstacle avoidance system [14].

2.3. Miscellaneous Techniques

According to [17], a different approach is taken, i.e., investigating the collision probability rate without considering temporal collision measurements as an intermediate or prerequisite quantity. In order to calculate the probability of colliding in a long duration of time, the crash probability rate must be integrated over time. Using the theory of surface intersections derived from vector stochastic processes, an upper bound is derived for collision probability rates. A saturated upper bound is a good approximation of collision probability rates under certain conditions. Despite being demonstrated on a surface with a rectangular shape and two dimensions with Gaussian distributions, the methodology may be used for any subset of surfaces that are smooth piecewise and multidimensional, even if they do not have a Gaussian distribution.

Monte Carlo simulations are used to obtain the real probability distribution of collision that can be approximated by the bound on the probability ratio derived from the simulations. Also included is an embedded platform package that is capable of calculating an estimate for the bounds associated with the rate of collision probability. For the scenarios considered in this paper, the distributions of collision probability rates obtained from this approximate formula are nearly similar to those calculated from mathematical methods. This method can be applied to specify the collision of a sample object with an extended object which is point-wise. For a collision of two circular-shaped extended objects or rectangular borders with the same fixed direction aligned to the axis, an extended object can also collide with a point-like object. The results showed that traditional distributions of collision time calculations, which are one dimensional, do not correctly describe collision statistics for two-dimensional environments and in the existence of environmental noise.

This study [18] examines crash safety models and objective definitions in general. In this paper, the validation issue using crash data is not addressed and indicators are not compared across different types of road infrastructure, but it takes a first step toward suggesting improvements for alternative safety analyses by comparing several different models and analyses.

Study [19] presents a methodological framework for calculating a temporary surrogate safety index based on sequential preemptive events within nonlinear traffic environments. The method used in this research provides an appropriate approach to calculating collision times in mixed nonlinear traffic scenarios with string traffic with a weak line, recognizing that the collision time index requires the integration of longitudinal and lateral descriptors.

According to a preliminary statistical analysis of car drivers following motorcycles, three-wheelers, cars, and trucks, collision times follow increasing centerlines and decrease on a decreasing trend. The collision time value for centerline separation is lower. Accordingly, the level of risk for drivers in the following scenarios varies based on the type of vehicle in front as well as the position of the vehicles on the road. The proposed method and recommended threshold values indicate that in the car chase mode, the severity of a collision is quite different for consecutive collisions. Therefore, in order to evaluate traffic safety in such cases, a comprehensive description of the minimum collision time thresholds should be provided for different side positions.

There are limitations to the work in this article. The probability that drivers have a lower minimum collision time threshold for car-following cases is assumed to be similar to the probability that drivers have a higher minimum collision time threshold for tailgating events. In the study, contrast severity was also estimated using the minimum recommended thresholds, which cannot be interpreted as an accurate outcome.

According to [20], it is possible to estimate collision times statistically. As a first step, it explained the mathematical basis for estimating the number of frames left until the robot hits the object. A color segmentation method is proposed for obstacle detection, and for object apparent size detection, it was shown that height or width can be used without using region growth. It is considered to be robust because it relies both on historical data and similar behavioral data. As a final step, the resulting data and models were analyzed statistically and compared in a quantitative and visual manner. By removing outlier data, the robustness of the framework has been investigated to create a more reliable model.

This study [21] proposes a new framework for estimating collision times in omnidirectional images for a flat surface with arbitrary orientation. The introduced method provides more information about the flat surface. Since gradient-based methods are simple, fast, and do not require high-level processing, they are a good option for real-time implementation. The results show that this approach can work effectively in complex real-world scenarios.

An algorithm for estimating TTC in synthetic sequences, stop-motion sequences, and real video is described and demonstrated [22]. With this method, features are not detected, tracked, or estimated, and latency is low. The range of operation can be extended to small collision times through spatial average and subsampling and to large collision times through temporal average and subsampling.

In [23,24], the authors used signal processing techniques for TTC estimation. In [25,26], the authors used potential field and cubic polynomials and an open-source software called CARLA for time-to-collision estimation.

In Table 1, we have classified different TTC approaches into groups of computer vision techniques, machine learning, deep learning, signal-based, video data-based, and other methods for consideration. Furthermore, the advantages and disadvantages of different methods used in the TTC estimation are presented.

In Figure 4, we have summarized the number of papers investigated in each approach used in TTC. As can be verified, most research work is oriented to computer vision-based approaches in TTC estimation.

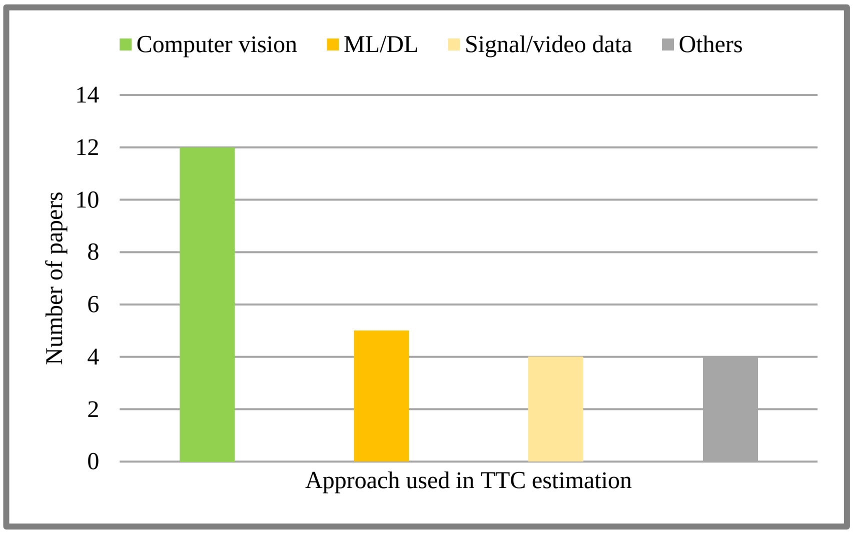

Figure 4. Paper distribution in different TTC approaches.

Table 1. Classification of the methods used in the studies related to TTC.

References	Computer Vision	Machine Learning	Deep Learning	Signal	Video Data	Others	Advantage	Disadvantage
[4–11]	√	-	-	-	-	-	Enhanced automation, improved accuracy, increased efficiency, improved accessibility	Limited context awareness, privacy/ethical concerns, bias and inaccuracy, dependence on infrastructure
[13,14]	√	-	√	-	-	-	Handling large and complex data, handling structured and unstructured data, improved performance	Increased complexity, overfitting tendencies, legal/ethical concerns
[15]	√	-	√	-	√	-	Handling large and complex data, handling structured and unstructured data, improved performance	Increased complexity, overfitting tendencies, legal/ethical concerns, dependency on video data
[12]	√	√	-	-	-	-	Handling large and complex data, automation, improved performance	Privacy/ethical concerns, dependence on infrastructure
[17,21,25,26]	-	-	-	-	-	√	Accurate mathematical modeling, enhanced prediction accuracy	Context dependent
[16]	-	√	-	-	-	-	Handling large and complex data, automation, improved performance	Privacy/ethical concerns
[19,23,24]	-	-	-	√	-	-	Using real-time sensory data, accuracy in prediction	Dependency on sensor/signal type

3. Collision Risk

The collision risk techniques evaluate the probability of collision in autonomous vehicles based on their current state and environmental conditions. The research work in this context can be categorized into three major domains, which are artificial intelligence (AI)-based techniques (supervised learning, unsupervised learning, reinforcement learning), sensor-based techniques, and other miscellaneous techniques that are mainly based on mathematical formulation, signal processing, and simulation tools.

3.1. AI-Based Techniques

In this study [27], the effectiveness of predicting driver behavior from real-time vehicle trajectory and time-to-collision data is investigated and evaluated. Calibrating the parameters of the optimal speed model for the study area based on distance and estimated TTC is accomplished using the virtual traffic (VT) lane framework to extract vehicle trajectories, motion classification, and speed estimates from the video. As a result of collision time-based model inference, improved deceleration was estimated for vehicles involved in car-following episodes and resulted in speed profiles with a calculated instantaneous speed error of 4.74 km/h and an instantaneous acceleration error of 19.84% less than the baseline error.

Modeling driving behavior in urban environments must take into account such characteristics as the differences between vehicles moving straight and vehicles turning at intersections. It has been shown that third-order polynomial functions are highly reliable in describing observed driving behavior (R2 of 70%), but video inference gives more information (including, but not limited to, signal state time, vehicle type, density of traffic, etc.).

As this study only tracks vehicles crossing intersections, the short length of vehicle trailing segments restricts its applicability. It is expected that vehicles following the car in the study area will engage in car-following behavior before or after the monitored section.

This study [28] introduces the C5.0 ensemble learning methodology as well as K-nearest neighbor, J48, and naive Bayes methods and a gradient boosting machine as an inclusive means for learning. As compared to the learner base and elastic network, the ensemble model is more accurate in predicting the target class. The outcome of driving is more predictable when sampling with replacement is used. Among the variables used to feed the models in this study are driver information, road information, and weather conditions. The C5.0 ensemble learning methodology, K-nearest neighbor, J48, naive Bayes, and gradient boosting machine are shown to be inclusive of these variables [29–31]. Ensemble models have a higher accuracy than learner bases and elastic networks when it comes to predicting the target class. The predictability of driving outcomes is increased by sampling with replacement. Variables selected to feed the models include driver, road, and weather data.

A proactive traffic safety management (PTSM) methodology is needed to deliver warning information in the vehicle, which is calculated based on estimating the collision risk in order to avoid accidents. The work in [30] has gathered the interconnection information between connected vehicles in Korea.

A sample vehicle-to-vehicle (V2V)-based proactive traffic safety control system is shown in Figure 5 (see [29]).

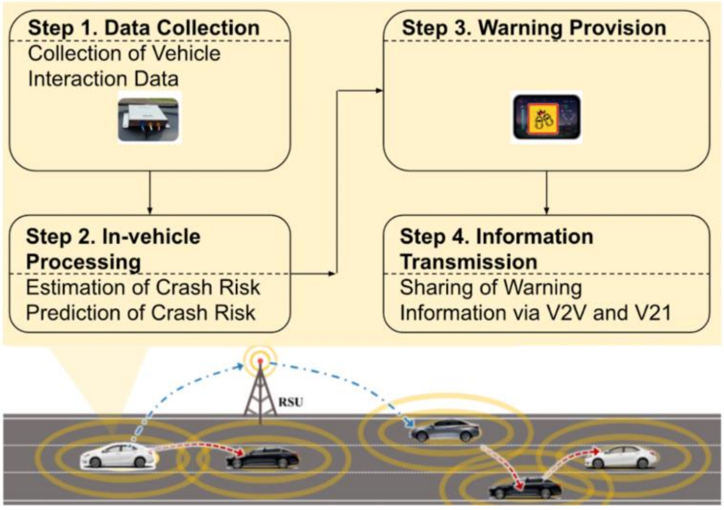
Step 1. Data Collection
Collection of Vehicle Interaction Data
Step 3. Warning Provision
Step 2. In-vehicle Processing
Estimation of Crash Risk
Prediction of Crash Risk
Step 4. Information Transmission
Sharing of Warning Information via V2V and V21
RSU

Figure 5. V2V-based proactive traffic safety control system [29].

Furthermore, this work uses V2V interactions to deliver an optimal threshold in order to provide warning data to be used in the vehicle. Nearly 2.16 million samples of data whose time of collision was less than 1.5 s have been gathered for 164 connected vehicles in 2019. The gathered information is in accordance with input and output information for a model that can predict the total collision risk. In order to propose estimation methods, the estimation and rolling time frames were calculated from 5 to 15 s (increment of 5 s) and from 1 to 3 s (increment of 1 s), respectively. Long short-term memory (LSTM) model meta-parameters were adjusted to 81 cases for unit numbers of hidden layers, optimal epoch numbers, small initial learning ratio, and batch size. Furthermore, the hyper-parameter of the K-nearest neighbor (KNN) method was fixed to 81 cases for the distance measure, weight of distance, and parameter K. Therefore, an accumulated number of 729 parametric cases were determined for every method. The optimized methods adjusted for meta-parameters and traveling salesman problems (TSP) predicted collision risk in 3 s with a mean absolute percentage error (MAPE) of nearly 8%. The warning data based on the estimated accident risk were shared between connected vehicles via V2V wireless links and adopted as early warning data for avoiding collisions.

Determining thresholds using errors of estimation is very vital to delivering efficient warning data in autonomous vehicles. The best threshold was obtained by calculating the detection failure rate (DFR), correct detection rate (CDR), and information provision rate (IPR), which are markers in predicting the reliability of warning data. Accident risk estimation errors are adjusted from 1% to 20%. Furthermore, threshold levels were adjusted from 0 to 1. The optimum threshold which has minimal IPR, complete CDR, and minimal DFR was calculated to be 0.69 when the estimation error is 8%. In addition, this threshold must be adjusted to be low in order to optimize the risk of missing samples as the estimation error rises. The proposed discussions proved that the mentioned policy is adopted to design a PTSM framework to deliver vehicle warnings based on collision risk estimation in connected vehicles.

At first, the result seems to be good, but more research efforts should be conducted to increase the validity of the developed policy. First, it is important to enhance the collision risk estimation models by investigating the connected vehicle behaviors before and after accidents. The properties of vehicle behavior are a function of the traffic collision impact area.

In addition, the accident risks of traffic flows may be estimated as a set of data obtained from the detection apparatus of the vehicle. In this scenario, it is important to leverage data obtained in a smaller time frame (for example, 10 Hertz) to exactly and instantly identify dangerous scenarios. Secondly, this work predefined dangerous cases as deliberate thresholds because the vehicle interaction information deviates from high-risk scenarios. In this scenario, the preadjusted threshold can cause some defects in the reliable delivery of warning data. As a bad scenario may be declared in multiple ways based on the accident time threshold, investigating the validity of warning data must be performed based on some sensitive analytical data using more advanced research. In the long term, it is recommended that the crash risk threshold be adjusted by monitoring the joint warning and collision data in the development phase of proposals.

Different safety metrics must be adopted for predicting the rear crash probability. Safety evaluation is available by calculating a minimal stop distance, which can be calculated based on the minimum stopping distance difference between the following and leading vehicles. Moreover, the implementation of different SSMs and collision risk estimation methods is adopted to efficiently represent the uncertain property of full risks.

Finally, the predicted collision risk in this work is a measure transformed by a probabilistic approach using accident time, and the observed collision time disturbance changes the estimation result. As the value of the threshold for delivering warning data can differ based on the size of the disturbance, more investigations on the disturbance prediction data gathering stage must be carried out in the future to enhance the presented strategy.

More studies must be performed on the design and implementation of those models that are capable of delivering optimized predictive results. Many other research works are required to enhance the mentioned policy and index to increase the validity of PTSM in connected vehicle (CV) settings.

The authors proposed a collision risk estimation method using LSTM [30] and K-nearest neighbor (KNN) models which are optimized based on the interaction information of vehicles by adjusting meta-parameters and TSPs.

In [31], sensory information, which is collected from self-driving cars, is consumed to deliver a complete risk evaluation strategy that is able to support all of the collision-related conditions. A methodology for risk inference is presented from artificial intelligence models (e.g., random forest and decision trees), which have been trained based on the Lyft dataset using proper feature engineering, and their estimations are delivered for multiple time frames in the future. Authors have proposed a machine learning-based method for describing the risk estimations of their presented models.

The proposed method can be incorporated into the safety mechanisms associated with autonomous cars. Furthermore, the method can be considered as a starting point for path estimation methods for road vehicles with high levels of risk to provide efficient maneuvers. The obtained models have a high performance (with an R2 score of 92.11 percent in a sample one-second regression scenario) and can be transferred and explained using more datasets (for example, the KITTI dataset).

In [32], a simple method to estimate the collision time is proposed to overcome the data loss challenge created by the small number of accidents in daily traffic. This research relies only on monocular cameras and uses depth and flow estimation methods based on deep learning to predict the vehicle speed using a simple and heuristic methodology, as well as object identification to detect the lead. The vehicle in question uses the approximate speed, boundary widths, and depth features of the single camera to estimate the collision time in the vehicle stopping positions, which is considered the moment of possible collision.

The developed algorithm was tested based on the automatic data captured from smart glasses and spherical cameras. The authors yielded a cross-validation root mean square error which is near one from the road viewpoint and self-oriented videos.

The advantages of adopting spherical cameras as a desirable alternative to several older cameras for visual vehicle tracking are also discussed.

In the research [33], a new method for collision risk assessment using deep prediction models has been presented. In particular, they have developed a special algorithm which is named Bayesian ConvLSTMs for spatio-temporal analysis of vision-related data, depth-sensing information, and driving commands to detect possible expected accidents. Unlike other deep learning methods in robotics, this method uses stochastic beliefs in the neural network output. These data can be processed for evaluating the forecasting uncertainties.

Many numerical evaluations show the importance of leveraging deep predictive models (DPM) in the evaluation of accident risks. In particular, the experiments show that it is feasible and beneficial to process the images of several cameras simultaneously in the same network architecture for this domain.

In work [34], the scientists try to detect accident risk and parking areas using deep learning (DL) fed from visual sensory data. A dataset was created in the parking area of Chungbuk National University to contrast it with a red–green–blue picture. The semantic segmentation method has been adopted for creating a network and merged with CSPNet to develop an optimal methodology that can be exploited with enhanced performance in a realistic vehicular scenario.

Many research efforts were carried out to optimize the activation and cost functions, in order to deliver optimal learning methods and accurate conditions. For employing the original pictures as they are, a special encoder with a low cost of computing was built. For optimizing the network associated with training, automatic mixed precision (AMP) was adopted to decrease the training time and memory, and also TensorRT was used to gain higher frames per second (FPS) data associated with the area of accident risk in outdoor and indoor scenarios.

To test the performance, the automatic RGB picture and the picture delivered from the NIIIS (National Institute of Intelligent Information Society) were evaluated, and a real car assessment was performed in the parking lot of Chungbuk University.

In the study [35], a test was conducted for forward accident alarms by a car with some visual and satellite navigation sensors. The test path was the Xian Rao Cheng Expressway. Some parameters such as distance between two following cars, speed, and position data were gathered at one Hertz frequency. A total of 1200 tracks were selected from the vehicle in vehicle chase mode. A collision time computational framework was developed for advanced driver-assistance systems (ADAS) and forward accident alarm systems. The comprehensive collision time model of urban highways comprises three sub-Gaussian distributions. A collision time of three seconds and speed were selected to be the input parameters for learning purposes. When the collision time is smaller than 2.97 s, the front collision warning is shown to the drivers.

Machine learning (ML) methodology for time-series processing, LSTM, and a deep belief network (DBN) were selected to characterize collision time. The data from the three-second segment were adopted to estimate the accident time in the next second.

In the research [36], five parameters of vehicle dynamics, the lateral and longitudinal movements of the vehicle on the test path, angle of front steering, deflection rate of lateral acceleration, and longitudinal speed, are evaluated. Vehicle movements in the lateral and longitudinal directions remain in a valid scope within the boundaries of the test path. The angle of front steering, yaw rate, and lateral acceleration are fixed to 0 during entering and before exiting the test path.

They achieved the lowest losses in the duration of two obstacle-prevention maneuvers. The controller managed to establish a suitable angle of front steering, while the car executed two obstacle prevention and lane change maneuvers and re-guided the vehicle to the target track in the middle lane before departing the assessment path. This management system is capable of creating a suitable angle of throttling alongside the assessment path.

At last, this tested algorithm for car dynamical systems and self-management policies showed to be an accurate tool for assessing dynamic car maneuvers on assessment paths, especially in early design stages. A TTC estimation framework using ML methods, which is presented in [36], has been shown in Figure 6.

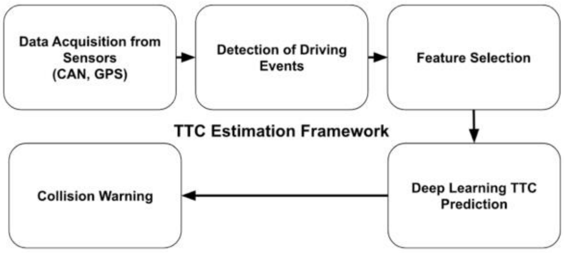
Data Acquisition from Sensors (CAN, GPS)
Detection of Driving Events
Feature Selection
TTC Estimation Framework
Deep Learning TTC Prediction
Collision Warning

Figure 6. A TTC estimation framework using ML techniques [36].

3.2. Sensor-Based Techniques

Jiang et al. [33] proposed a collision risk prediction for vehicles with sensor data fusion through a machine learning pipeline. They have shown that the combination of multiple sensors (e.g., LiDAR, RADAR, monocular, and binocular cameras) promotes the perception and decision-making process of vehicles.

In the paper [37], laser scan data were used for automatic accident prevention. The mentioned work presents the practical guides related to implementing accident prevention mechanisms and benefits from traditional work findings related to obstacle prevention and detection scenarios. The proposed accident prevention mechanism has been designed, developed, and tested. Choosing the best performance and managing the stimulus in each situation, this new system considers steering maneuvers (including braking, etc.) to prevent collisions and find obstacle-free areas.

In the proposed work, a laser scanner with one layer is adopted for detecting obstacles, and a satellite-based positioning system working on a digital map delivers some important data that the car should move to avoid a collision/accident. However, this understanding mechanism is not perfect enough to guarantee that a particular steering maneuver is safe for other vehicles.

A detection frequency of 10 Hz has been used for the laser scanner. Although this extent of detection frequency is appropriate for many scenarios, higher values are suggested in scenarios with high mobility and scenario change levels.

Enhanced methods detect obstacles and calculate collision time. Moreover, the decision-making methods must be enhanced for use in newer algorithms to consider complicated driving cases in order to have a proper response in every scenario, which consists of a mixed steering–braking management strategy.

At last, the proposed fuzzy logic manager forces more use cases. As an example, complicated systems that interact with this simple fuzzy logic (FL) controller may only deploy simple control and speed messages for their response regulations.

The work in [38] simply uses the motion of a linear feature set to calculate the collision time, which is basically usable in all environments and prevents complex vehicle search and detection using video data. Selected areas for spatio-temporal motion categorizing yield dangerous accident alarms and reduce the computing overhead for real-time analysis. This algorithm is pioneering research that only uses motion information and has been evaluated for different vehicle videos. Avoiding the detection of the vehicle makes this method useless for the shape and type of the target vehicle.

In [39], sensor data from autonomous vehicles are used in order to present a complete risk evaluation which includes all of the different conditions that may lead to a collision in order to assess the risk associated with it. A method for inferring risk from Lyft data using machine learning methods (random forests and decision trees) with estimations performed over multiple time frames is presented. The risk predictions of models can be explained using a tree-based explanatory technique. Vehicles that are self-driving can incorporate this directly into their safety systems. Also, it can serve as the initial point for path estimation methods for those road users with a high level of risk, and also be used to help the vehicle in navigating the roads safely. It is possible to transfer and explain the models obtained on other datasets (like the KITTI data) and they are highly effective (with an R2 score of 92.11% for a regression sample of 1 s).

In this work, a novel deep learning (DL) methodology based on direct perception for self-driving vehicles is introduced. Different from previous work that paid attention to the potential feature engineering of a given convolutional neural network (CNN), for the first time, the performance of self-driving cars has been investigated for multiple DL scenarios.

Furthermore, the developed algorithm is not limited by incorrect prejudices about other cars around the self-driving vehicle. As an example, the framework runs without any assumptions regarding the speed of cars. Instead, the methodology adopts a more accurate model with the assumption of more sensor availability that provides the distance between vehicles surrounding the self-driving car.

The behavior of three popular convolutional neural networks in extracting the features of the road has been evaluated. The findings show that GoogLeNet is the most exact network for these scenarios. Apart from evaluating the performance of the features, they have also proposed the use of more features for assessing the self-driving vehicle's performance.

At last, the proposed features are used to differentiate this methodology in comparison with traditional ones, and the performance of this system is clearly enhanced compared to traditional works, which try to navigate the entire path. The cause of this enhancement is the elimination of overlapping parameters.

Authors in [40] cover various aspects of safety for intelligent components that use machine learning techniques to integrate artificial intelligence for self-driving. The attention was on the principal problems for guaranteeing safety in important scenarios that use artificial intelligence algorithms, especially neural networks. Older safety methods are not correctly tailored for these systems; hence, it is necessary to develop more comprehensive tools such as monitoring methods. The presented collision estimation method will deliver a satisfactory extent of safety for the required systematic operations.

In another approach by [41], gathering a sufficient number of samples, in the context of object tracking noise, for approximation of collision probability can help reduce such incidents. In this work, at each time step, the controller type selection relies on opting for the minimum probability of collision. This method, however, could lead to some aggressive actions taken by the controller. To increase the safety measures, factors such as analyzing the distance to the surrounding vehicles, as well as utilizing a probability field to compute a path around potential collision areas, were suggested.

The versatility of this control system for self-driving cars was highlighted as it was not tailored to a specific traffic situation or intersection, needing only an environment map with prespecified routes on the environment map. However, simulations reveal potential challenges with the required noise level of this approach, as it solely uses a global positioning system (GPS) and communication for object tracking; to overcome such challenges, the use of sensors was recommended.

Centered on predicting the trajectory of each car, a method was introduced in [42] to assess the risk of collisions, using GPS error data and shape characterization. In this approach, so-called high-level Kalman filters with a set of precise dynamic models and risk indicators were utilized to improve the performance of risk estimation, trajectory tracking, and describing the severity of potential accidents.

In another work outlined in [43], deep neural networks as well as particle filter-based systems were used to detect and track observable objects in traffic. In this real-time approach, the distance and relative speed of surrounding vehicles were estimated via basic principle modeling to assess the expected risks up to ten seconds ahead of time during the course of driving the car.

In a study highlighted by researchers [44], simulated data emerge as a valuable resource, especially when the available dataset is not sufficient for constructing a robust predictive model. Mid-horizon forecasting faces challenges due to the limited sample size attributed to the complexity of high-dimensional behavioral features essential for learning forecasting models.

To overcome this hurdle, domain-matching methods leverage simulated data to enhance predictions in both synthetic and real-world scenarios. This finding signifies the potential for enhancing medium-horizon risk prediction systems through offline training of predictive models using simulated data.

The proposed approach holds the potential for refinement through advanced modeling, estimation, and forecasting techniques. Leveraging driver models learned from data could notably enhance transmission performance by more accurately capturing human failure patterns.

Employing a multilinear approach to importance sampling facilitates the transfer of real-life collision prediction, enhancing system usability. To improve prediction performance, employing domain-matching models that explicitly delineate shared and unique latent feature spaces proves advantageous. However, the local maximum likelihood estimation approach for inferring behavioral parameters in next-generation simulation (NGSIM) vehicles exhibits limitations, particularly in the inability to represent individual driving behaviors and generalize across diverse drivers.

Risk assessment analysis was extensively studied in [45–47]. The risk assessment of failure in autonomous vehicles due to malfunctions in sensing devices was studied in [45]. Based on this analysis, sensors such as LiDAR, GPS, RADAR, and wheel encoders collectively exhibit a probability of failure exceeding 20 percent. The methodology of this risk assessment was performed via a tree-based analysis of critical events.

In [46], the effectiveness of required braking and safety measures are studied via conditional random fields utilized to assess various driving style preferences. However, it is important to note that the study acknowledges a limitation, as it does not benefit from cloud traffic data. In [47], mitigation functions, such as impact point, as well as the dependence on numerical–moral aspects, were considered to assess integrated interaction strategies where regulated comfort safety measures were desirable.

Accordingly, the detection of accidents in traffic flows [48] and identification of dangerous locations on the road using smartphone data [49] were investigated. To detect accidents, one approach is to use multitask adaptation [48]. In this approach, the adaptation of context relation as well as appearance, compatibility measurements, and motion between consecutive frames were utilized to detect various accidents and non-involved scenarios. The results of this method were tested against DADA-2000 and AnAn (A3D) datasets. In the method proposed by [48], a hybrid thresholding method consisting of EuroFOT and adaptive speed dependency for high-risk driving scenarios was used to distinguish between driving styles within a predefined speed range. The knowledge base system starts with EuroFOT thresholding and gradually uses the driver speed-dependent thresholding to improve the accuracy; this eventually delivers the inference to a map that traffic and safety experts could use to improve the system and mitigate potential hazards.

The authors in [49] have used self-supervised consistency learning for a collision risk assessment. Aichinger et al. [50] have adopted low-cost smartphone sensor data for locating crash risk spots in a road network.

In [51], the authors proposed a direct vehicle collision detection algorithm from motion in driving videos. Meng et al. have proposed a vehicle trajectory prediction method based on predictive collision risk assessments for autonomous driving in highway scenarios [52].

The authors in [53] used reinforcement learning for predictive trajectory planning of autonomous vehicles at intersections. In [54], a new integrated collision risk assessment methodology for autonomous vehicles has been proposed.

The authors in [55] developed an advanced collision risk model for autonomous vehicles.

3.3. Miscellaneous Techniques

According to a theory of the risk field, the risk assessment index of an undercarriage process creates risk avoidance based on the field of risk theory. In the work [56], loop detector data have been merged with accidental data to develop a simulation model to estimate pre-accident track data. A hazard avoidance index is calculated based on the trajectory data. A comparison of the indicator's performance in crash and non-crash modes is performed to determine its validity. The result is that this indicator can significantly distinguish between accident states and non-accident states. In comparison with collision time and deceleration rate to avoid collision (DRAC), the risk aversion indicator is found to be superior, since it includes more parameters that affect the collision risk, such as space-related safety and time advance. Additionally, the risk avoidance indicator can be used to assess the risk faced by the self-driving car during the car chase process. As a final step, the work proposes a model that is risk aversion index-based and binary for predicting accident risk. In comparison with other loop detector data-based collision estimation models, its AUC (area under the ROC curve) value reaches 0.96.

Simulated route data are used in this paper. Considering that the simulation model is idealized compared to the actual traffic environment, there are some limitations that occur in the simulation process due to the idealization of the model. It is difficult to accurately reproduce the pre-crash trajectory when more regular vehicle behavior is compared to the behavior in the real world. As well, the simulation scene that is discussed in this paper is all carried out in free-flow mode and does not discuss the more complex traffic flow environment that is used in the real world.

In [57], analytical solutions for calculating collision state probability (CSP) and collision event probability (CEP) are presented. Based on comparisons with Monte Carlo simulations, both solutions provide numerically accurate results. The performance of these simulations is 100 to 800 times better than that of MC simulations. As a result of this work, it has been demonstrated that it is possible to calculate, in real time, the risk from a variety of possible moves in a scene, which can be performed by all traffic participants in the scene. As a result, the self-driving vehicle planner will not only have valuable information at their disposal, but they may also have the opportunity to provide risk-based attention control in the future as well.

Analytical methodologies for deriving CSP and CEP are proposed in the work [57]. Both of the proposed methods have been proven to give accurate results in comparison with Monte Carlo numerical analysis. In fact, the performance of the methods is between 100 and 800 times superior to its Monte Carlo counterpart. In this study, it is also deduced that it is possible to calculate, in real time, the risks associated with multiple maneuvers of the participants in the traffic of the mentioned scenario.

This fact suggests that effective data for the designers of autonomous driving vehicles are necessary and also may be used for the attention-based management of collision risks in future research.

The study [58] introduces the smart data integration apparatus for accident alarm estimation for avoiding the two imminent challenges of driving risk and inaccurate accident alarm because of noises in the input signal in the time interval of pre-accident alarm analysis.

More-than-visual sensing, GPS, and drowsiness of steering are roughly applied to the additional response time when tuning the presented NHTSA method, along with a strange fault-tolerant approach to accommodate imprecise input signals. The authors present a QTBPNN/ANFISA method included in a dual-distributed device to develop accurate and correct event data logging and alerting. Many experiments were successfully conducted and showed that the presented algorithm is superior to the two popular accident alarm methodologies.

In addition, anomaly detection in runtime safety monitoring and AI-based solutions, usage of historical accident assessment to avoid hazards on the road, and location-specific motion detection in car camera video were widely studied in the research studies by [59–61], respectively. However, in [60], root cause analysis of collisions and black spot detection were performed via statistical distribution analysis and sliding window methods to formulate preventative control actions; in [61], Bayesian inference of arbitrary motions in the in-car video feed was used to detect potential dangers on the road.

Hortel et al. have used a formal model for collision risk assessment of AVs [62]. In the work [63], a potential risk assessment for the safe driving of autonomous vehicles under occluded vision has been investigated.

Song et al. have proposed a driving and steering collision avoidance system for autonomous vehicles with model predictive control based on non-convex optimization [64].

In Table 2, we have classified different collision risk estimation approaches into groups of computer vision techniques, machine learning, deep learning, simulation-based, sensor-based, video data-based, LiDAR-based, time-series data-based, and other methods for consideration. Moreover, in Table 2, we have described the advantages/disadvantages of different collision risk assessment methods.

In Figure 7, we have summarized the number of papers investigated in each approach deployed in collision risk assessment. As can be verified, AI-based methods have the most significant attention in collision risk assessments.

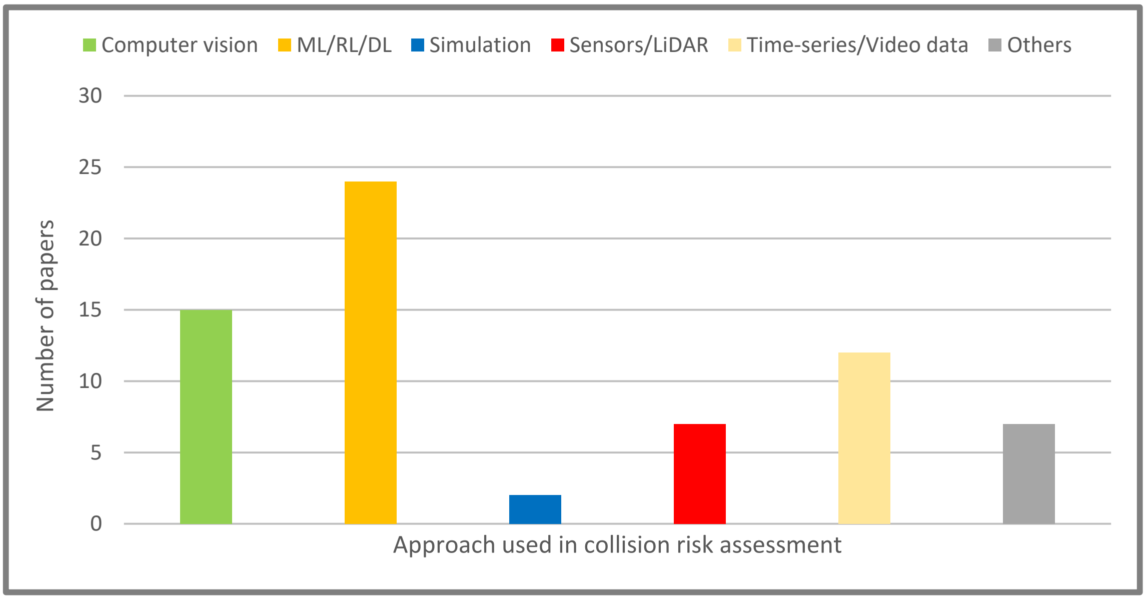

Figure 7. Paper distribution in different collision risk assessment approaches.

Table 2. Classification of the methods used in the studies related to Collision Risk.

References	Computer Vision	ML	DL	Simulation	Sensor	Video Data	Lidar	Time-Series Data	RL	Others	Advantage	Disadvantage
[46,47,56,58,62–64]	-	-	-	-	-	-	-	-	-	√	Accurate mathematical modeling, enhanced prediction accuracy	Context dependent
[27,30–32,34,51]	√	-	√	-	-	√	-	-	-	-	Handling large and complex data, handling structured and unstructured data, improved performance	Increased complexity, overfitting tendencies, legal/ethical concerns, dependency on video data
[29,59]	-	-	√	-	-	-	-	-	-	-	Handling large and complex data, handling structured and unstructured data, improved performance	Increased complexity, overfitting tendencies, legal/ethical concerns
[57]	√	-	-	√	-	-	√	-	-	-	Enhanced automation, improved accuracy, increased efficiency, improved accessibility, higher precision, more flexibility due to simulations	Limited context awareness, privacy/ethical concerns, bias and inaccuracy, dependence on infrastructure, bias on LiDAR data
[28,33,49,52–55]	-	√	-	-	-	-	-	-	-	-	Handling large and complex data, automation, improved performance	Privacy/ethical concerns
[37,60]	√	-	-	-	-	-	-	-	-	-	Enhanced automation, improved accuracy, increased efficiency, improved accessibility	Limited context awareness, privacy/ethical concerns, bias and inaccuracy, dependence on infrastructure
[36]	√	√	-	-	-	-	-	√	√	-	Handling large and complex data, automation, improved performance, accurate mathematical modeling, model-free, online	Privacy/ethical concerns, dependence on infrastructure
[41]	-	√	√	-	√	-	-	-	-	-	Handling large and complex data, automation, improved performance, using exact sensory data	Privacy/ethical concerns, increased complexity, overfitting tendencies, bias on sensor type
[44]	√	-	√	-	-	√	-	√	-	-	Handling large and complex data, handling structured and unstructured data, improved performance, accurate mathematical modeling	Increased complexity, overfitting tendencies, legal/ethical concerns, dependency on video data
[38]	√	-	-	-	-	√	-	-	-	-	Enhanced automation, improved accuracy, increased efficiency, improved accessibility	Limited context awareness, privacy/ethical concerns, bias and inaccuracy, dependence on infrastructure, bias on video data
[50]	-	√	-	-	√	-	-	-	-	-	Handling large and complex data, automation, improved performance, using exact sensory data	Privacy/ethical concerns, bias on sensor type
[39,40]	√	-	√	-	√	-	-	-	-	-	Handling large and complex data, handling structured and unstructured data, improved performance, using exact sensory data	Increased complexity, overfitting tendencies, legal/ethical concerns, bias on sensor type
[35]	√	-	√	-	-	-	-	√	-	-	Handling large and complex data, handling structured and unstructured data, improved performance, accurate mathematical modeling	Increased complexity, overfitting tendencies, legal/ethical concerns
[61]	-	-	-	-	-	-	-	√	-	-	Accurate mathematical modeling, improved estimation accuracy	Inefficient in complex scenarios

Table 2. *Cont.*

References	Computer Vision	ML	DL	Simulation	Sensor	Video Data	Lidar	Time-Series Data	RL	Others	Advantage	Disadvantage
[43]	-	-	-	-	√	-	-	-	-	-	Using exact sensory data, reduced complexity	Bias on sensor type, decreased performance
[42]	-	-	-	-	√	-	√	-	-	-	Using exact sensory data, reduced complexity, higher precision	Bias on LiDAR data, decreased performance
[45]	-	-	√	√	-	-	-	-	-	-	Handling large and complex data, handling structured and unstructured data, improved performance, more flexibility due to simulations	Increased complexity, overfitting tendencies, legal/ethical concerns
[48]	-	√	-	-	-	√	-	-	-	-	Handling large and complex data, automation, improved performance	Privacy/ethical concerns, bias on video data

4. Data and Methods

In this chapter, we will investigate the data and methods which have been adopted in the previous two chapters (i.e., TTC estimation and collision risk assessment). Please note Tables 3 and 4. Refs. [65–134] are about the used datasets in the paper.

Table 3. Data and methods used in TTC.

Reference	Data	Method
[4]	Captured sequence of frames involving displacement change.	Active Contour Affine Scale (ACAS), with image flow approximated by an affine transformation
[5]	Captured Sequence of images at a constant vehicle speed.	Featureless Based Control with Kalman Filtering and Gain Scheduling
[6]	Captured ten sets of sequence images with five scenes.	Acceleration constraint (τ-constraint) and distance constraint (Φ-constraint) methods
[7]	Feature position information of two consecutive sets of image sequences as well as odometry data.	Feature Based Robust TTC Calculation
[8]	Captured Video on an Android Smartphone	Relative Velocity Estimation via Depth and Motion Sensing
[9]	Captured Sequence of Images from a Light Source via a Fixed Camera	Analyzing Photometric Features via Measuring Changes in Intensity with Ambient Elimination
[10]	Captured radar sensor data labeled by hand	Spot detection and paring
[11]	Synthetic object data	Using visual stimulus degradation
[12]	Synthetic and Real Sequence of Images with constant velocity camera	Dense Optical Flow-Based Time-to-Collision
[13]	FlyingChairs2 [65] and FlyingThings3D Datasets [66]	Binary, Quantized, and Continuous Estimation
[14]	EVReflex Dataset	Merged event camera and lidar streams without requiring prior scene geometry or obstacle knowledge.
[15]	KITTI Vision Benchmark [67]	Horizontal and Vertical Motion Divergence
[16]	Synthetic Event-Based Camera Model using DVS	Vertical Descent Control via Estimating Divergence of Optical Flow
[17]	Monte Carlo Simulation	Linking the collision probability rate distribution to TTC distribution and deriving the upper bound for the collision distribution.
[18]	Video feeds from 1 to 3 merging zones were captured across 20 roundabouts in Quebec, covering a total of 37 distinct sites.	Assessing TTC Indicators and Aggregation Methods using Constant Velocity, Normal Adaptation, and Motion Pattern Prediction Methods for Surrogate Safety Analysis.
[19]	video feeds from five Indian cities with varied lane setups, straight road sections away from intersections, in clear weather conditions and visibility.	Use urban road trajectory data to create temporal safety indicators in non-lane-based traffic
[20]	Captured video of 210 frames on a moving monocular vision robot	Estimate TTC using the so-called Tau-margin by the ratio of change of apparent size of obstacles via color segmentation
[21]	Real and Synthetic Images by Pov-ray Software [68]	Spatial and Temporal Gradient-based methods
[22]	Synthetic Video Feed and Stop-Motion Sequence	Featureless Direct Method via Brightness Derivative Based on Two Consecutive Frames as a MontiVision Filter
[23]	Synthetic Data Using AirSim [69] Simulator, and Aerial Drones	Multi-object Deep Feature Detection via Pixel-Level Object Segmentation
[24]	UDACITY Sensor Fusion Dataset [70]	Detector Descriptor TTC Detection
[25]	MATLAB/Simulink Simulation	Time To Collision Lane Change via Potential Field and Cubic Polynomial
[26]	PointCloud2 Data via CARLA Simulator [71]	Bypass Object Detection with Line Intersection via Laser-Scan

Table 4. Data and methods used in collision risk assessment.

Reference	Data	Method
[27]	UA-Detrac Dataset: Traffic Video Feed Data with Over 1.2 million Bounding Boxes [72]	Data-Driven Optimal Velocity Model via Deep-SORT Tracker, Classification by VT-Lane Framework, and Kalman Filtering
[28]	VTTI Dataset [73]	Ensemble Model of C5.0, K-Nearest Neighbor, J48 Classification, Naive Bayes, and Gradient Boosting Machine
[29]	Vehicle Interaction Data	Proactive V2V-based Warning System via Long Short-Term Memory (LSTM) Risk Prediction
[31]	Lyft Level-5 [73], KITTI Vision [67] Benchmark, and Waymo [110] datasets	Planar 2D Collision Machine Learning Model via Decision-Trees and Random-Forests
[32]	Self-Collected Video Sequence Similar to KITTI Vision Benchmark [67]	Monocular Vision Deep Learning via Optical Flow Modeling, Depth Estimation, Object Detection, Ego-vehicle Speed Estimation, Lead Vehicle Identification, and Car Stop Collision Timing
[33]	NuScenes Dataset [106]	Multi-sensor Fusion to Extract Motion Features and Predict Trajectories, via Minimum Future Spacing (MFS) and Extended Kalman Filter (EKF)
[34]	Synthetic Data via Robotics Simulator CoppeliaSim (V-REP) [75]	Bayesian Convolutional Long Short-Term Memory (LSTM)
[35]	Around View Monitor (AVM) Images with Four 190 Wide-Angle Cameras Based on the Chungbuk National University Parking Lot, and National Institute of Intelligent Information Society (NIA) Dataset	Bird-Eye View Area Detection and Semantic Segmentation via CSPHarDNet, CSPDenseNet, and HarDNet
[36]	Car-Following On-board Diagnosis (OBD) Dataset Collected from Xi'an Rao Cheng Expressway for Three Days	Forward Collision Detection in Advanced Driver-Assistance Systems (ADAS) via Long Short-Term Memory (LSTM) and Deep Belief Network (DBN)
[37]	Single-Layer Laser-Scanner Data with 10 Hz Frequency	Automatic Collision Avoidance System via Obstacle Detection and Fuzzy-Logic Control
[38]	Forward Facing Camera Data Based on TASI 110-Car Naturalistic Driving Study [76], Caltech Pedestrian, ETH, TUD-Brussels, Daimler, and INRIA Datasets	Motion Divergence Detection by Analyzing Horizontal and Vertical Trace Expansion form Cluster of Line Segments
[39]	TORCS (The Open Racing Car Simulator)	Direct Perception Deep Learning via GoogLeNet Autonomous Driving (GLAD) and ConvNet based on AlexNet
[40]	Event Data Recording via GPS and Vision Sensing, and Vehicle-to-Vehicle Communication	Neural Network (QT-BPNN) and Adaptive Network-Based Fuzzy Inference System (ANFIS), with Distributed Dual-Platform DaVinci+XScale_NAV270
[41]	Synthetic Data Using Automated Test Trajectory Generation (ATTG)	Bayesian Deep Learning (BDL), and Reinforcement Learning
[42]	LIDAR, Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I), GPS, and IMU Readings	Robust System to Estimate the Likelihood of Collisions, Validated by Controlling the Ego Velocity of Vehicle with Velocity Planning Control (VPC)
[43]	Two La Route Automatise (LaRA) vehicles exchanging their positions and velocities, and GPS Data	Trajectory Prediction and Kalman Filtering via Geometric and Dynamic Approaches
[44]	KITTI Dataset [67]	FasterRCNN-101 for Object Detection, Particle-based Object Tracking and Distance Estimation Based on First Principles, and Risk Assessment Based on Inverse Time-to-Collision (TTC), using Monocular Vision Input Feed.
[45]	Next-Generation Simulation (NGSIM) Dataset [77]	Generate High-risk Scenes Using a Bayesian Network Model.
[46]	California Department of Motor Vehicles Autonomous Vehicle Testing Records [78–82], Virginia Department of Transportation (DOT) [83], the New York State DOT Traffic Crash Reports [84], NASA Risk Assessment Data [85]	Fault Tree Analysis: Bayesian Belief Network for LIDAR and Camera Failure, Chi-square distribution for Radar Failure, Extended Markov Bayesian Network for Software Failure, Kalman filter for Wheel Encoder Failure, Least Squares for GPS Failure, Generic quorum-system evaluator for Database System Failure, In IEEE 802.11b Network and CAP for Communication Failure, Markov Chain Model for Integrated Platform Failure, Human Reliability Analysis (THERP, CREAM, and NARA) for Human Command Error, and Artificial Neural Networks on Clean Speech for System Failure for Human Command Detection

Table 4. *Cont.*

Reference	Data	Method
[47]	CARLA Simulator	Risk Assessment Based Decision-Making to Avoid Collisions with Likelihood Analysis via Conditional Random Field (CRF)
[48]	Captured Video Feed	Bayesian Framework and Decision Trees
[49]	AnAn Accident Detection (A3D) [111], and DADA-2000 [86] Datasets	Appearance, Motion, and Context Consistency Learning via Self-Supervised Consistency Traffic Accident Detection (SSC-TAD) Learning
[50]	Collected 200 Hours of Smartphone GPS and Video Data	Quantile Regression (QR) Modeling, EuroFOT Thresholding, and Spacial Clustering
[51]	Collected 30 Frames-per-second (FPS) Video Data	Analyzing Horizontal and Vertical Motion Divergence without Object Detection and Depth Sensing
[52]	NGSIM US101 [77], and highD [87] Datasets	Long Short-term Memory (LSTM) Encoder-Decoder for Sequence Generation, Convolutional Social Pooling (CSP) for Extracting Local Spatial vehicle interactions, and Graph Attention Network (GAN) for Distant Spatial Vehicle Interactions
[53]	Lyft Level-5 Dataset [74]	Reinforcement Learning: Partially Observable Markov Decision Process (POMDP) for Sequential Decision-making, Bayesian Gaussian Mixture Models for Learning Patterns of Trajectory, and Gibbs Sampling for Validating Simulations
[54]	Athens Dataset [88]	Dynamic Bayesian Networks (DBN)
[55]	UK [89], and Athens [88] Datasets	Network-level Collision Prediction (NLCP), and Dynamic Bayesian Networks (DBN)
[56]	NGSIM [77], and California's Caltrans PeMS System Loop Detector [90] Datasets	Collision Risk Assessment Indicator by Risk Repulsion
[57]	Monte Carlo Simulation	Collision state probability (CSP), and Collision event probability (CEP)
[58]	ISO 3888 Test Track [91]	Dynamical Autonomous Intelligent Vehicles (AIV) Modeling and Fuzzy Logic Control
[59]	Automotive Data and Time Triggered Framework (ADTF) [92]	Risk-based Criticality Measurement: General Integral Criticality Measurement, Integration of Severity Prediction Functions, Environmental Risk Parameters, and Multi-dimensional Risk Mdoels
[60]	SLAM System [93]	Run-time Safety Monitoring Framework via Lane Detection, and Object Detection
[61]	Hungarian Road Network from the Hungarian Central Statistics Department (HCSD)	Historical Accident Data Risk Assessment Based on Thresholding and Distribution Analysis via Sliding Window
[62]	CARLA Simulator [71]	Formal Conformance Test Generation, and Statistical Analysis on Traces
[63]	Dynamic Simulation	Occluded Vision Analysis Based on Dynamic Bayesian Network (DBN) Inference
[64]	Constant Turn-Rate and Acceleration (CTRA)	Micro-genetic Algorithm, and Model Predictive Control (MPC)
[65]	Vehicle-to-Everything (V2X) Data	Directed Acyclic Graph (DAG) Blockchains
[66]	LIBRE [99], Foggy Cityscape [100], CADCD [101], Berkley DeepDrive [102], Mapillary [103], EuroCity[104], Oxford RobotCar[105], nuScenes [106], D2-City [107], DDD17 [108], Argoverse [109], Waymo Open [110], A3D [111], Snowy Driving [112], ApolloScape [113], SYNTHIA [114], P.F.B [115], ALSD [116], ACDC [117], NCLT [118], 4Seasons [119], Raincouver[120], WildDash [121], KAIST multispectral [122], DENSE [123], A2D2 [124], SoilingNet [125], Radiate [126], EU [127], HSI-Drive [128], WADS [129], and Boreas [130] Datasets; as well as, CARLA [71], LG SVL [94], dSPACE [95], CarSim [96], TASS PreScan [97], AirSim [69], and PTV Vissim [98] Simulation Environments	Assessing Effects of Weather on Automated Driving Systems (ADS) Perception and Sensing

Table 4. *Cont.*

Reference	Data	Method
[67]	BLE AoA Dataset [131,132]	Analyze Indoor Localization Systems: Wireless Fidelity (Wi-Fi), Ultra-Wide Bandwidth Radio (UWB), Inertial Measurement Units(IMU), Bluetooth Low Energy (BLE),
[68]	Long Range Wide Area Network (LoRaWAN) [133,134]	Vehicle Geolocalization

5. Open Areas

Research [4] indicates that optimizing collision time involves applying smoothing techniques over time, such as the Kalman filter. Implementing recursive least squares (RLS) with a forgetting parameter yields satisfactory outcomes. Although such filtering methods enhance accuracy, they might introduce delays in collision time estimation, posing challenges in real-time navigation. In pursuit of a scalable approach, a pixel pipeline architecture incorporates collision time estimation algorithms compatible with hardware. This setup allows for flexible adjustments between estimation accuracy, processing speed, and resource utilization. Additionally, some researchers such as those in [5] aim to expand and evaluate a smart video sensor in forthcoming studies.

In study [6], a comprehensive collision avoidance strategy is introduced. In this system, the collision decision-making process was performed using point estimators or their time integrals. Based on another recent study highlighted in [7], the results of aerial robot landing control and navigation can be applied to miniature robots solely by vision sensors, expanding the applicability of the findings. Addressing the transferability of safety indicators, ref. [9] emphasizes the importance of comparing results across datasets of various road infrastructure types, extending beyond roundabouts to include highways, intersections, collector roads, and more. This approach enhances the broader applicability of safety indicators.

The use of the central limit theorem is discussed in relation to model reliability in [12]. Despite the potential for increased reliability with a larger dataset, ref. [12] demonstrates that results with smaller datasets can be as efficient and reliable. However, augmenting the dataset has the potential to yield more acceptable results. Building upon the work presented in [14], future directions for extension are considered. One possibility is the expansion of the study to encompass general movement along the X and Y axes, indicating the potential for broader applications and insights in the research domain.

In research cited as [18], it was discovered that vibrations affecting the car-mounted camera led to significant image blurring, which negatively impacted the accuracy of collision time estimation. To counteract this issue, the smartphone gyroscope sensor was used to measure camera rotational speed. This offers a solution to mitigate image movement during collision time calculation.

Following the insights from study [20], upcoming investigations would concentrate on employing alternating lights as a potential solution for addressing pairing issues, as well as increasing imaging resolution to enhance accuracy. Moreover, to formulate a predictive model for traffic flow instability, the approach outlined in [27] suggested analyzing the index distribution before the accident. In addition, ref. [28] highlighted the need to encompass longer arterial segments and assess the applicability of the proposed model in calibrating specific driving behaviors, i.e., the potential to act as an external driver model for microscopic simulations.

According to research [29], enhancing prediction accuracy in accidents and near-accident situations may be achievable by considering a comprehensive set of factors, including driver and vehicle information as well as driver time-series datasets in various environmental conditions. This holistic approach could contribute to more robust predictive models. Continuing on the same line, further exploration in [32] aims to investigate whether analyzing the temporal and spatial distribution of CEP can serve as additional planning input. In the context of decision making, ref. [34] suggests the need for improvement to consider complex driving styles and situations to ensure steering and braking control are integrated into the solutions and the decision-making process for a more comprehensive and responsive approach to autonomous vehicle control.

Researchers in [37] express the intention to incorporate additional parameters, such as driver error tolerance, into their computational model to proactively avoid accidents. In addition, the research reviewed in [38] highlights the need for investigating and applying better utilization of existing proprioceptive data for enhanced autonomous vehicle performance. Some studies, for example [39], suggest exploring special parking areas, such as women-only or lots for the disabled, to address the gaps in assessing the risk of collisions in parking lots. They proposed conducting research to improve performance and develop algorithms for collision risk identification and precise distance measurement using camera and LiDAR technologies.

In the pursuit of deeper evaluations of the dynamic behavior of autonomous vehicles [41], the recommendation is to integrate dynamic systems into the car model, including brake systems, steering, and suspension. The call for implementing different independent control strategies and comparing their performance across various benchmark vehicle test tracks emphasizes the importance of practical assessments to evaluate the performance of autonomous vehicle control systems.

In ref. [44], the researchers suggested the upcoming endeavors would focus on embedded real-time system design, delving deeper into the subsystems introduced in the paper and investigating interactions between multiple vehicles. This suggests a commitment to practical application and a more comprehensive understanding of system interactions. According to [45], future research plans would be dedicated to exploring more complex behaviors including but not limited to left or right turns at intersections across the track and similar intricate traffic scenarios.

The authors of the abovementioned real-time object tracking approach [46] identified the limitations of the design for investigation and debugging in future work, including the environment itself and the dependence on specific parameters for state and speed estimation. While the absence of consideration for traffic mass information in the method derivation described in [47] was highlighted as an area for future improvement and development, the authors of [50] mentioned a more robust dynamic mapping strategy for weight selection in the objective function and prediction of crash severity to evaluate the overall performance as the main areas for future improvement.

For future research in [51], the aim is to employ online anomaly detection to enhance the framework's dynamic safety and expand to create safety profiles in the design time phase. In [52], the future direction involves expanding the scope with methods such as sliding windows in two dimensions or other relevant data mining procedures to implement alternative black spot search approaches.

The authors of [53] proposed further modifications to risk classification with more accurate assessments of real-world datasets to elevate the model's applicability in complex situations such as curved case scenarios. As highlighted in [55], the forthcoming focus is to scale up the applications of their presented method to bigger dataset cases and longer-term exploratory datasets to handle more realistic scenarios.

Furthermore, blockchain and mmWave with THz are going to be employed for V2X communications. Blockchain technology may be capable of securing V2X data, as well as enabling other useful V2X services such as payments [135]. On the other hand, mmWave with Thz is going to replace LiDAR for inclement weather [136].

For safety considerations, multiple sensors and technologies will be employed. Different technologies like UWB, BLE, and long-range wide-area networks (LoRAWANs) will be used for geolocalizing vehicles usually in indoor areas. This exact localization results in identifying the exact location of the vehicles in such areas and so reduces the risk of collision enormously [137,138].

In Table 5, the main open area fields with their related references are listed.

Table 5. Categorizing the main open area fields.

Open Area Field	Related References
Real-time navigation	[4,5]
Using novel statistical methods	[6,7,12,27,28,50]
Generalization	[9,14,41,44,45,53,55]
Using new sensors	[18,20]
Enhancing robustness	[29,32,34]
Parameter engineering	[37–39,46,47]
Safety considerations	[51,137,138]
Deploying novel technology (blockchain and mmWave, etc.)	[52,135,136]

6. Challenges

The development of a fully autonomous vehicle introduces several challenges. Managing numerous variables simultaneously within driving performance is critical, as even a single error could lead to a catastrophic accident. Typically, human intervention guides basic system-level movements and decisions. Despite substantial advancements in processing speed, sensor quality, and camera technology, these systems still harbor weaknesses that hinder their widespread public release and universal adoption. The rest of this section discusses several aspects of the challenges in autonomous vehicle design, i.e., calculations, system design, implementation, generalization, validation, and safety considerations.

6.1. Calculations

One of the main challenges of deep learning methods, also as outlined in [39], lies in the requirement for substantial data and time for network training, contributing to prolonged training periods and increased costs in autonomous vehicle design. To address this issue, a common strategy is to integrate reinforcement learning with supervised learning, aiming to reduce training time and enhance the final output.

However, the necessity for extensive data collection remains a formidable obstacle in training autonomous vehicles to navigate diverse real-world scenarios. Collaborative data sharing among companies could expedite the transition from experimental to commercial systems, yet due to market competition, data-sharing reluctance persists. Increasing the volume of available data in autonomous system design is beneficial for handling more complex conditions, but it introduces the challenge of data diversity.

When training data are abundant, there is a risk of overfitting the model if diversity is not considered. This could lead to a situation where the system performs well only on the trained data, which highlights the importance of data variety in minimizing overfitting issues. Thus as datasets grow, attention to diverse data types becomes crucial to avoid overfitting problems [27,29,32–35,39].

The computational complexity in autonomous systems is primarily influenced by the states and decisions agents need to make. As systems must continuously make decisions, the increase in dimensions leads to exponential growth in computational complexity, termed dimensionality perturbation. In high-dimensional cases, this significantly impacts solution complexity. Discretization can also be useful, but it may compromise system performance. Alternate approaches involve employing multiple learning agents, utilizing evolutionary strategies for parallelization, or removing unnecessary data from input and training datasets. The challenge extends to the deployment phase, where the high computational load of artificial neural networks poses hurdles.

This challenge affects both network development/training and the on-vehicle computing system. Deep learning algorithms demand high computing power, leading to increased power consumption, a critical consideration in system design. The interplay between computational demands, power consumption, and system design underscores the complexities faced in the development and deployment of autonomous systems [39].

6.2. System Design

One significant challenge associated with deep learning revolves around the intricate task of selecting the neural network architecture. There are no established guidelines for determining an optimal neural network structure tailored to a specific solution. If the network is too small, both in terms of size and layer count, it results in suboptimal system performance. On the other hand, employing too many neurons poses the risk of overfitting the training data, and the surge in computational complexity further complicates matters [2].

Beyond architecture, various parameters, such as the learning rates, training procedure, determination of batch sizes, and objective function, significantly influence any agent's performance and convergence. The process of selecting these parameters lacks well-defined methods, often resorting to trial and error due to the intricate architecture of artificial neural networks. This involves tuning a wide range of hyper-parameter values and optimizing each based on performance, a process hindered by the computational demands of training sessions [27,29,32–35].

To address this challenge, ongoing research explores alternative methods for parameter selection, including automatic approaches that seek optimal values within a specific spectrum or through model-based methods. Different techniques, such as grid search methods, coordinate reduction, network search, random search, and neural architecture search, are being investigated to streamline this process.

Coordinate reduction, for instance, involves keeping all hyper-parameters fixed except one, allowing for the systematic determination of the optimal value for each parameter individually. Another approach, network search, relies on sheer computing power to explore all possible states without employing specialized intelligence to enhance efficiency [44,51,59].

However, this method's computational cost escalated significantly when dealing with a large number of neural network models, making it practical only when models can be trained swiftly. In contrast, random search offers a faster alternative to grid search by randomly sampling within the specified parameter interval. Despite its efficiency, a drawback lies in the incomplete coverage of the parameter space, leading to closely situated sample points.

This limitation can be mitigated by employing pseudo-random sequences. Recent advancements in neural architecture search methods consider hardware efficiency by analyzing hardware feedback on the learning signal. This innovative approach has resulted in the development of neural network architectures optimized for specific hardware platforms, showcasing a tangible performance advantage. Importantly, these methods hold the potential to extend their applicability to discover efficient network architectures tailored for automotive hardware platforms, aligning with the growing emphasis on specialized solutions in the field of autonomous systems.

Another challenge is that current continuous driving control systems often rely on smaller networks and datasets. As datasets grow, the potential for employing deeper architectures emerges, offering the promise of reduced bias and variance in training and consequently more robust control policies. The quest for specialized network architectures for autonomous driving, exemplified by models like conditional imitation learning, represents a focal point for future research endeavors in this domain. Ultimately, the goal is to design architectures tailored to the unique demands of autonomous vehicles, thereby advancing the field of self-driving technology.

6.3. Implementation

In the implementation of reinforcement learning systems, setting appropriate goals poses a distinct challenge. One notable advantage of reinforcement learning lies in the absence of a need to implicitly define the agent's behavior, a characteristic found in rule-based systems. Instead, only the reward function, often more straightforwardly defined than the objective function, and the control function for braking, steering, and more must be specified in a way to maximize the long-term cumulative reward. Consequently, accurately capturing the desired behavior of the agent within this reward function is crucial [36]; otherwise, unexpected and undesirable behavior may manifest. To address this, the concept of reward shaping is introduced, wherein intermediate rewards are utilized to guide the agent toward the desired behavior, deviating from the binary rewards typically used for success or failure.

In scenarios where agents control both lateral and longitudinal conditions, a further challenge arises in defining a reward function when the agent is required to execute multiple actions, such as braking, steering, and acceleration. Agents rely on feedback from the reward function to enhance their performance. However, complications arise when multiple actions are involved, making it unclear which action led to a specific reward. One solution is to utilize a hybrid reward architecture, where the system employs a decomposed reward function for various scenarios [36].

Ensuring the integrity of the reward function in reinforcement learning systems is paramount for system developers, as misuse can lead to unforeseen and undesirable behavior by the agents, commonly referred to as a bonus hack. This phenomenon, known as reward hacking, occurs when the agent discovers unanticipated ways to exploit the reward function for substantial rewards, contrary to the intended goals. To counteract reward hacking, adversarial reward functions are introduced, resembling generative adversarial networks, wherein the reward function becomes a factor itself. This allows the agent to explore the environment and fortify its strength for reward hacking.

An alternative approach is employing a look-ahead model that evaluates rewards based on state predictions. Implementing reward-limiting strategies offers a straightforward solution by imposing a maximum value on the reward function, curbing the potential for unexpected high-reward scenarios. Additionally, employing multiple reward functions proves advantageous, as the complexity of hacking multiple rewards is typically higher than manipulating a single reward. In addressing the challenges of goal setting and reward function manipulation, an alternative strategy involves leveraging inverse reinforcement learning to extract the reward function from specialized task representations.

6.4. Generalization

Addressing diverse environments poses a significant challenge for learned control systems, particularly in achieving scalability. A driving strategy effective in an urban setting may not be optimal on a highway, given the distinct traffic patterns and safety considerations. Similar challenges arise with changing weather conditions and seasons. The neural network's ability to apply knowledge from prior experiences to navigate entirely new environments is termed generalization. However, the challenge lies in ensuring that even if the system demonstrates effective generalization in one new environment, it can extend this capability to a range of potential environments.

Due to the intricate operating conditions of vehicles, comprehensive testing is impractical, making it challenging to build and validate deep learning systems that can generalize across diverse situations. As autonomous vehicles must adapt to varied usage environments, overcoming this challenge is crucial for the real-world deployment of deep learning-based autonomous vehicles. To prevent poor generalization in deep neural networks, it is essential to stop training before overfitting occurs [27,29,32–35,39,44,51,59]. Overfitting happens when the model fits the training data so closely that it loses its ability to generalize to new data. Determining the optimal stopping point to avoid overfitting remains an unsolved problem. However, employing three different datasets—training, validation, and test sets—provides a strategy for assessing generalizability.

The validation set helps mitigate overfitting by monitoring errors. If a decline in accuracy in the validation set occurs during training iterations, indicating an increase in network error, training should be stopped to prevent overfitting. This approach ensures that changes reducing error in the training set are aligned with a decrease in error in the validation set, preserving the network's ability to generalize effectively.

Beyond preventing overfitting, a validation set serves as a means to compare various network architectures, such as networks with differing hidden layer numbers. Consequently, an additional independent set, termed the test set, becomes imperative to assess the performance of networks without biases. This test suite rigorously evaluates the final network's capabilities without influencing network or architecture selection.

To improve testing accuracy while sometimes increasing training errors, regularization techniques are employed. A range of regularization techniques exists, including L1 and L2 regularization, which add extra values to the model's objective function, compelling the network to favor smaller weights. This minimizes internal noise impact, allowing the network to learn broader trends across the dataset. Weight truncation, another method, involves imposing constraints directly on network weights, demonstrating improvements in model robustness. Dropout, a widely used approach, randomly removes neurons during training, updating only the remaining weights. This dynamic process prevents intricate neuron adaptations, aiding each neuron in focusing on task-relevant features, thus minimizing overfitting tendencies.

6.5. Validation

Ensuring the accuracy of system performance and security verification is crucial, yet real-world experiments come with considerable time and financial expenses. Consequently, large-scale vehicle studies often rely on collaborations between government research projects and manufacturers, mitigating costs and complexities. Simulation studies emerge as cost-effective, faster, and more flexible alternatives, serving as an initial step to assess performance and safety. Simulations offer the unique advantage of replicating scenarios challenging to recreate in reality, like accidents, and have become a dominant method owing to the enhanced accuracy and speed of simulation tools [45,57].

However, despite simulations' benefits, verifying and validating model errors during the simulation process remains critical. Inaccuracies in modeling intricate mechanical interactions, such as joints and friction, can significantly impact real-world system performance. Over-reliance on simulated environments without proper adaptation to the real world presents challenges in effectively transferring policies learned within simulations. For systems intended for real-world use, a combination of field testing and simulation-based training is imperative.

While simulation aids in the convergence of reinforcement learning algorithms due to the multitude of required experiments, ensuring successful policy transfer from simulator to reality is vital. Recent research in robotic arms demonstrates promising policy transfer from simulators to real-world scenarios. Validating the model and simulated environment alone may not be sufficient for self-driving vehicles. Emphasis should also be placed on validating the quality of training data. Ensuring datasets accurately represent the intended operational environments and encompass potential scenarios is crucial. Biased datasets favoring specific actions, such as leaning towards turning to a specific direction, or scenarios, e.g., daytime driving, might introduce detrimental biases into the learning model, prompting the need for robust dataset validation to detect biases or harmful patterns that could lead to undesirable behavior in learned policies [45].

6.6. Safety Considerations

Safety measures in autonomous vehicles are very critical, and any malfunction may lead to severe consequences. As neural networks grow more intricate, understanding their solutions becomes increasingly complex, often leading to a "black box problem". While these systems may perform well in a validation environment, testing them across all real-world scenarios remains impractical [8,27,29,31,41,59,60].

Without a clear grasp of the decision-making process within these systems, ensuring safe decisions in new environments becomes challenging, especially in online learning methods where policies evolve dynamically, potentially transitioning from safe to unsafe procedures over time. In autonomous vehicle systems, reacting safely to unpredictable behaviors of other vehicles or pedestrians is crucial. Incorporating unsafe driving behaviors of other vehicles in the training data can equip the vehicle controller to handle such situations. Enhancing reliability and safety involves methods like damage memory, storing rare negative events for continual training reinforcement, and ensuring safe behaviors.

During initial reinforcement learning training, ensuring safety is crucial. Especially during initiation phases, when agents take heuristic-based learning approaches in environments involving road users or pedestrians, inappropriate actions due to exploration can lead to catastrophic outcomes [31].

In the pursuit of solutions, several approaches are explored. One involves using in real-life (IRL)-like slots to showcase safe behavior examples, serving as a reference point for baseline safety. Simulated exploration within controlled environments aids in detecting potential issues, yet its scalability for real-time systems is limited, posing challenges. Similar safety constraints apply to system testing and evaluation, necessitating exhaustive precautions until the proper and secure operation is assured. Deep neural networks exhibit vulnerabilities to physical adversarial samples, posing significant implications for real-world applications [59,60].

The subtle alterations in hostile samples, imperceptible to the human eye, could make it difficult to defend against such attacks. These weaknesses raise security concerns across various technologies leveraging deep reinforcement learning. While defenses are proposed, advanced attacks can circumvent these measures, showcasing the persistent challenges in safeguarding against adversarial attacks in deep learning applications.

7. Conclusions

This research delves into time to collision and collision risk management in autonomous vehicles, exploring strategies like predicting collision times and implementing alarms to prevent obstacles. The analysis aims to categorize diverse techniques, encompassing image processing, machine learning, deep learning, and sensors, among others, to address collision risks. Additionally, the paper outlines challenge areas (including safety considerations, validation, generalization, implementation, system design, and calculations) and potential future research directions within this critical domain.

The references collectively highlight various novel methods and their comparative performance in estimating time-to-collision (TTC) and obstacle avoidance accuracy. Ref. [4] demonstrates ACAS's superiority over SIRS and IBD in TTC estimation, highlighting the specific strengths of each method. The proposed method in [5] outperforms optic flow-based techniques in speed and accuracy. Similarly, ref. [6] showcases significant accuracy and speed improvements compared to visual-inertial odometry methods. Different methods presented in refs. [7–14] demonstrate enhancements in performance, stability conditions for control systems, robustness without specific requirements, improved detection rates, and resolution of individual modality failures during obstacle avoidance. Ref. [15] indicates high accuracy and precision for the proposed method. However, limitations exist, such as the simulator in [16] not meeting stringent safety standards for space missions despite promising outputs. Additionally, ref. [17] emphasizes the need for more comprehensive collision avoidance systems beyond classical TTC distributions, especially in two-dimensional geometries with process noise.

The discussed references highlight various methodologies and technologies in the domain of autonomous vehicles and collision risk assessment. Ref. [18] emphasizes consistent measurement methodologies, while [19] employs multiple minimum TTC criteria for traffic safety assessment. Ref. [20] focuses on color segmentation for obstacle identification, and [21] utilizes gradient-based methods for simplicity and speed in real-world scenarios. Refs. [22,23] detail methods with minimal latency and adaptability to maritime and traffic monitoring contexts, respectively. Ref. [24] ranks TTC estimation methods, favoring camera-derived distance calculations. Ref. [25] introduces TTCA-LC for safer lane changes, ref. [27] emphasizes diverse behaviors at intersections, and [28] showcases an ensemble model for accuracy using driver, roadway, and weather data. Methods in [29,31–33] present crash risk prediction, high-performance risk assessments, video evaluation, and multi-sensor fusion algorithms for collision forecasting, respectively. Refs. [34–39] discuss deep predictive models, deep learning algorithms, TTC models, collision avoidance systems, and fault tolerance in detection methods.

Refs. [40–46] cover advanced collision warning systems, crash prediction networks, collision probability estimation, risk assessment via GPS-based representations, and fault tree analysis for autonomous vehicle failure. The collection of references covers a wide spectrum of methodologies and technologies pertinent to collision risk assessment and autonomous vehicle safety. These references explore diverse approaches such as risk assessment modules using conditional random fields [47], Bayesian rules for risk identification [48], and self-supervised consistency learning in dashcam videos [49]. Other methods involve adapting thresholds based on speed-related information [50], filtering motion traces for time-to-collision accuracy [51], and predicting collision risks between autonomous and other vehicles based on trajectories [52].

Additionally, the references discuss reinforcement learning-based trajectory planning [53], machine learning classifiers for collision prediction [54], and incorporating simulation models for conflict-based autonomous vehicle risk assessment [55]. Methods for predicting unstable traffic flow [56], analytic solutions for computing collision indicators [57], and nested approaches for assessing autonomous vehicle behaviors [58] are also presented. Some references focus on the ethical and numerical aspects of risk evaluation [59], dynamic safety enhancements using anomaly detection techniques [60], and alternative techniques for self-driven vehicles [61]. The use of formal verification tools for perception components in AVs [62], challenges in adverse weather conditions [63], and comprehensive control approaches for collision avoidance [64] are explored as well. Further areas of exploration involve the necessity for robust V2X blockchains [135], challenges related to LiDAR safety and imaging costs [136], methods for localization adaptability [137], as well as geolocalization approaches discussed in [138].

Author Contributions: Conceptualization, P.G.; methodology, P.G.; validation, P.G. and B.H.; investigation, P.G. and B.H.; resources, P.G.; writing—original draft preparation, P.G. and B.H.; writing—review and editing, P.G. and B.H.; visualization, P.G. and B.H.; supervision, P.G.; project administration, P.G.; funding acquisition, P.G. and B.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Datasets regarding this paper are listed in references [65–134].

Acknowledgments: The authors must thank Zargari, M. Ahmadi and H. Amiri for helping in creating draft version of paper and also must appreciate anonymous reviewers for their insightful comments.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

5G NR	Fifth-Generation New Radio
ACAS	Active Contour Affine Scale
ADAS	Advanced Driver-Assistance Systems
ACV	Autonomous Connected Vehicle
AI	Artificial Intelligence
AMP	Automatic Mixed Precision
AUC	Area Under the ROC Curve
AV	Autonomous Vehicle
BLE	Bluetooth Low Energy

CDR	Correct Detection Rate
CEP	Collision Event Probability
CSP	Collision State Probability
C-V2X	Cellular Vehicle to Everything
CV	Connected Vehicle
DBN	Deep Belief Network
DFR	Detection Failure Rate
DL	Deep Learning
DRAC	Deceleration Rate to Avoid Collision
DSRC	Dedicated Short-Range Communication
FL	Fuzzy Logic
FPS	Frames Per Second
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HD	High Definition
IBD	Image Brightness Derivative
IPR	Information Provision Rate
KNN	K-Nearest Neighbor
LiDAR	Light Detection And Ranging
LSTM	Long Short-Term Memory
MAPE	Mean Absolute Percentage Error
MEC	Multi-Access Edge Computing
ML	Machine Learning
NGSIM	Next-Generation Simulation
NFV	Network Function Virtualization
PTSM	Proactive Traffic Safety Management
RADAR	Radio Detection And Ranging
RGB	Red–Green–Blue
ROC	Receiver Operating Characteristic
RL	Reinforcement Learning
SDN	Software-Defined Networks
SIRS	Scale-Invariant Ridge Segment
TSP	Traveling Salesman Problems
TTC	Time-To-Collision
TTI	Time-To-Impact
V2V	Vehicle-to-Vehicle
VIO	Visual Inertial Odometry
VT	Virtual Traffic
UWB	Ultra-Wide Bandwidth

References

1. Ahangar, M.N.; Ahmed, Q.Z.; Khan, F.A.; Hafeez, M. A Survey of Autonomous Vehicles: Enabling Communication Technologies and Challenges. *Sensors* **2021**, *21*, 706. [CrossRef]
2. Hakak, S.; Gadekallu, T.R.; Reddy, K.R.; Swarna, M.; Ramu, P.; Parimala, M.; De Alwis, C.; Liyanage, M. Autonomous vehicles in 5G and beyond: A survey. *Veh. Commun.* **2023**, *39*, 100551. [CrossRef]
3. Chen, L.; Li, Y.; Huang, C.; Xing, Y.; Tian, D.; Li, L.; Hu, Z.; Teng, S.; Lv, C.; Wang, J.; et al. Milestones in Autonomous Driving and Intelligent Vehicles—Part I: Control, Computing System Design, Communication, HD Map, Testing, and Human Behaviors. *IEEE Trans. Syst. Man Cybern. Syst.* **2023**, *53*, 5831–5847. [CrossRef]
4. Alenya, G.; Negre, A.; Crowley, J.L. A comparison of three methods for measure of time to contact. *IEEE/RSJ Int. Conf. Intell. Robot. Syst. IROS* **2009**, *2009*, 4565–4570. [CrossRef]
5. Zhang, H.; Zhao, J. Bio-inspired vision based robot control using featureless estimations of time-to-contact. *Bioinspir. Biomim.* **2017**, *12*, 025001. [CrossRef]
6. Burner, L.; Sanket, N.J.; Fermuller, C.; Aloimonos, Y. Fast Active Monocular Distance Estimation from Time-to-Contact. 2022. Available online: http://arxiv.org/abs/2203.07530 (accessed on 20 December 2023).
7. Sagrebin, M.; Pauli, J. Improved time-to-contact estimation by using information from image sequences. *Inform. Aktuell* **2009**, *2009*, 26–32.
8. Wang, L.; Horn, B.K.P. Time-To-Contact control for safety and reliability of self-driving cars. *Int. Smart Cities Conf. ISC2* **2017**, *3*, 16–19. [CrossRef]

9. Watanabe, Y.; Sakaue, F.; Sato, J. Time-to-contact from image intensity. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 4176–4183. [CrossRef]
10. Gormer, S.; Muller, D.; Hold, S.; Meuter, M.; Kummert, A. Vehicle recognition and TTC estimation at night based on spotlight pairing. In Proceedings of the 2009 12th International IEEE Conference on Intelligent Transportation Systems, St. Louis, MO, USA, 4–7 October 2009; pp. 196–201. [CrossRef]
11. Hecht, H.; Brendel, E.; Wessels, M.; Bernhard, C. Estimating time-to-contact when vision is impaired. *Sci. Rep.* **2021**, *11*, 21213. [CrossRef]
12. Shi, C.; Dong, Z.; Pundlik, S.; Luo, G. A hardware-friendly optical flow-based time-to-collision estimation algorithm. *Sensors* **2019**, *19*, 807. [CrossRef] [PubMed]
13. Badki, A.; Gallo, O.; Kautz, J.; Sen, P. Binary TTC:A temporal geofence for autonomous navigation. *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* **2021**, *2021*, 12941–12950. [CrossRef]
14. Walters, C.; Hadfield, S. EVReflex: Dense Time-to-Impact Prediction for Event-based Obstacle Avoidance. *IEEE Int. Conf. Intell. Robot. Syst.* **2021**, *2021*, 1304–1309. [CrossRef]
15. Kilicarslan, M.; Zheng, J.Y. Predict Vehicle Collision by TTC from Motion Using a Single Video Camera. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 522–533. [CrossRef]
16. Sikorski, O.; Izzo, D.; Meoni, G. Event-based spacecraft landing using time-to-contact. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Nashville, TN, USA, 19–25 June 2021; pp. 1941–1950. [CrossRef]
17. Altendorfer, R.; Wilkmann, C. A new approach to estimate the collision probability for automotive applications. *Automatica* **2021**, *127*, 109497. [CrossRef]
18. St-Aubin, P.; Saunier, N.; Miranda-Moreno, L. Comparison of Various Time-to-Collision Prediction and Aggregation Methods for Surrogate Safety Analysis. *Transp. Res. Board 94th Annu. Meet.* **2015**, *1*, 1–21.
19. Das, S.; Maurya, A.K. Defining Time-to-Collision Thresholds by the Type of Lead Vehicle in Non-Lane-Based Traffic Environments. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 4972–4982. [CrossRef]
20. Sanchez Garcia, A.J.; Rios Figueroa, H.V.; Hernandez, A.M.; Cortes Verdin, M.K.; Vega, G.C. Estimation of time-to-contact from Tau-margin and statistical analysis of behavior. In Proceedings of the 2016 International Conference on Systems, Signals and Image Processing (IWSSIP), Bratislava, Slovakia, 23–25 May 2016; pp. 1–6. [CrossRef]
21. Benamar, F.Z.; El Fkihi, S.; Demonceaux, C.; Mouad-dib, E.; Aboutajdine, D. Gradient-Based time to contact on paracatadioptric camera. In Proceedings of the IEEE International Conference on Image Processing, ICIP'2013, Melbourne, Australia, 15–18 September 2013.
22. Horn, B.K.P.; Fang, Y.; Masaki, I. Time to contact relative to a planar surface. *IEEE Intell. Veh. Symp. Proc.* **2007**, *2007*, 68–74. [CrossRef]
23. Tottrup, D.; Skovgaard, S.L.; Sejerson, J.F.; Figueiredo, R.P. Real-Time Method for Time-to-Collision Estimation from Aerial Images. *J. Imaging* **2022**, *8*, 62. [CrossRef]
24. Ozbek, M.; Celebi, A.T. Performance Evaluation of Camera-Based Time to Collision Calculation with Different Detectors & Descriptors. *Eur. J. Sci. Technol.* **2021**, *32*, 59–67.
25. Lin, P.; Javanmardi, E.; Tao, Y.; Chauhan, V.; Nakazato, J.; Tsukada, M. Time-to-Collision-Aware Lane-Change Strategy Based on Potential Field and Cubic Polynomial for Autonomous Vehicles. *arXiv* **2023**. [CrossRef]
26. Beyrle, M. *Time To Collision Calculation for an Autonomous Model Vehicle with CARLA*; Technical Reports in Computing Science; University of Applied Sciences: Kempten, Germany, 2020.
27. Abdelhalim, A.; Abbas, M. A Real-Time Safety-Based Optimal Velocity Model. *IEEE Open J. Intell. Transp. Syst.* **2022**, *3*, 165–175. [CrossRef]
28. Bugusa, Y.; Patil, S. An improved accident crash risk prediction model based on driving outcomes using ensemble of prediction algorithms. *Int. J. Sci. Technol. Res.* **2019**, *8*, 603–611.
29. Jo, Y.; Jang, J.; Ko, J.; Oh, C. An In-Vehicle Warning Information Provision Strategy for V2V-Based Proactive Traffic Safety Management. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 19387–19398. [CrossRef]
30. Staudemeyer, R.C.; Morris, E.R. A tutorial into Long Short-Term Memory Recurrent Neural Networks. *Ralf C. Staudemeye* **2019**, *2019*, 1–42.
31. Nahata, R.; Omeiza, D.; Howard, R.; Kunze, L. Assessing and Explaining Collision Risk in Dynamic Environments for Autonomous Driving Safety. *IEEE Conf. Intell. Transp. Syst. Proc. ITSC* **2021**, *2021*, 223–230. [CrossRef]
32. Rill, R.A.; Farago, K.B. Collision Avoidance Using Deep Learning Based Monocular Vision. *SN Comput. Sci.* **2021**, *2*, 1–10. [CrossRef]
33. Jiang, Y.; Hu, J.; Liu, H. Collision Risk Prediction for Vehicles with Sensor Data Fusion through a Machine Learning Pipeline. In Proceedings of the International Conference on Transportation and Development, Seattle, WA, USA, 31 May–3 June 2022.
34. Strickland, M.; Fainekos, G.; Ben-Amor, H. Deep predictive models for collision risk assessment in autonomous driving. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 4685–4692. [CrossRef]
35. Lee, S.; Lee, D.; Kee, S.C. Deep-Learning-Based Parking Area and Collision Risk Area Detection Using AVM in Autonomous Parking Situation. *Sensors* **2022**, *22*, 1986. [CrossRef]

36. Guo, L.; Jia, Y.; Hu, X.; Dong, F. Forwarding Collision Assessment with the Localization Information Using the Machine Learning Method. *J. Adv. Transp.* **2022**, *2022*, 9530793. [CrossRef]
37. Jimenez, F.; Naranjo, J.E.; Gomez, O. Autonomous collision avoidance system based on accurate knowledge of the vehicle surroundings. *IET Intell. Transp. Syst.* **2015**, *9*, 105–117. [CrossRef]
38. Kilicarslan, M.; Zheng, J.Y. Bridge motion to collision alarming using driving video. *Proc. Int. Conf. Pattern Recognit.* **2016**, *2016*, 1870–1875. [CrossRef]
39. Al-Qizwini, M.; Barjasteh, I.; Al-Qassab, H.; Radha, H. Deep learning algorithm for autonomous driving using GoogLeNet. In Proceedings of the 2017 IEEE Intelligent Vehicles Symposium (IV), Los Angeles, CA, USA, 11–14 June 2017; pp. 89–96. [CrossRef]
40. Chang, B.R.; Tsai, H.F.; Young, C.P. Intelligent data fusion system for predicting vehicle collision warning using vision/GPS sensing. *Expert Syst. Appl.* **2010**, *37*, 2439–2450. [CrossRef]
41. Nair, S.; Shafaei, S.; Kugele, S.; Osman, M.H.; Knoll, A. Monitoring safety of autonomous vehicles with crash prediction network. *CEUR Workshop Proc.* **2019**, *2301*.
42. Annell, S.; Gratner, A.; Svensson, L. Probabilistic collision estimation system for autonomous vehicles. In Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil, 1–4 November 2016; pp. 473–478. [CrossRef]
43. Ammoun, S.; Nashashibi, F. Real time trajectory prediction for collision risk estimation between vehicles. In Proceedings of the 2009 IEEE 5th International Conference on Intelligent Computer Communication and Processing, Cluj-Napoca, Romania, 27–29 August 2009; pp. 417–422. [CrossRef]
44. Phillips, D.J.; Aragon, J.C.; Roychowdhury, A.; Madigan, R.; Chintakindi, S.; Kochenderfer, M.J. Real-time Prediction of Automotive Collision Risk from Monocular Video. 2019. Available online: http://arxiv.org/abs/1902.01293 (accessed on 20 December 2023).
45. Wulfe, B.; Hartong-Redden, R.; Chintakindi, S.; Kodali, A.; Choi, S.C.T.; Kochenderfer, M.J. Real-time prediction of intermediate-horizon automotive collision risk. *Proc. Int. Jt. Conf. Auton. Agents Multiagent Syst. AAMAS* **2018**, *2*, 1087–1096.
46. Bhavsar, P.; Das, P.; Paugh, M.; Dey, K.; Chowdhury, M. Risk analysis of autonomous vehicles in mixed traffic streams. *Transp. Res. Rec.* **2017**, *2625*, 51–61. [CrossRef]
47. Li, G. Risk assessment based collision avoidance decision-making for autonomous vehicles in multi-scenarios. *Transp. Res. Part C. Emerg. Technol.* **2021**, *122*, 102820. [CrossRef]
48. Kilicarslan, M.; Zheng, J.Y. Towards collision alarming based on visual motion. In Proceedings of the 2012 15th International IEEE Conference on Intelligent Transportation Systems, Anchorage, AK, USA, 16–19 September 2012; pp. 654–659. [CrossRef]
49. Fang, J.; Qiao, J.; Bai, J.; Yu, H.; Xue, J. Traffic Accident Detection via Self-Supervised Consistency Learning in Driving Scenarios. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 9601–9614. [CrossRef]
50. Aichinger, C.; Nitsche, P.; Stutz, R.; Harnisch, M. Using Low-cost Smartphone Sensor Data for Locating Crash Risk Spots in a Road Network. *Transp. Res. Procedia* **2016**, *14*, 2015–2024. [CrossRef]
51. Kilicarslan, M.; Zheng, J.Y. Direct vehicle collision detection from motion in driving video. In Proceedings of the 2017 IEEE Intelligent Vehicles Symposium (IV), Los Angeles, CA, USA, 11–14 June 2017; pp. 1558–1564. [CrossRef]
52. Meng, D.; Xiao, W.; Zhang, L.; Zhang, Z.; Liu, Z. Vehicle Trajectory Prediction based Predictive Collision Risk Assessment for Autonomous Driving in Highway Scenarios. *arXiv* **2023**, arXiv:2304.05610.
53. Zhang, E.; Zhang, R.; Masoud, N. Predictive trajectory planning for autonomous vehicles at intersections using reinforcement learning. *Trans. Research Part C Emerg. Tech.* **2023**, *149*, 104063. [CrossRef]
54. Katrakazas, C.; Quddus, M.; Chen, W.-H. A new integrated collision risk assessment methodology for autonomous vehicles. *Accid. Anal. Prev.* **2019**, *127*, 61–79. [CrossRef]
55. Katrakazas, C. Developing an Advanced Collision Risk Model for Autonomous Vehicles. Ph.D. Dissertation, Loughborough University Research Repository, Loughborough, UK, 2017.
56. Wu, B.; Yan, Y.; Ni, D.; Li, L. A longitudinal car-following risk assessment model based on risk field theory for autonomous vehicles. *Int. J. Transp. Sci. Technol.* **2021**, *10*, 60–68. [CrossRef]
57. Philipp, A.; Goehring, D. Analytic collision risk calculation for autonomous vehicle navigation. *Proc. IEEE Int. Conf. Robot. Autom.* **2019**, *2019*, 1744–1750. [CrossRef]
58. Sabry, Y.; Aly, M.; Oraby, W.; El-demerdash, S. Fuzzy Control of Autonomous Intelligent Vehicles for Collision Avoidance Using Integrated Dynamics. *SAE Int. J. Passeng. Cars Mech. Syst.* **2018**, *11*, 5–21. [CrossRef]
59. Hruschka, C.M.; Topfer, D.; Zug, S. Risk Assessment for Integral Safety in Automated Driving. In Proceedings of the 2019 2nd International Conference on Intelligent Autonomous Systems (ICoIAS), Singapore, 28 February–2 March 2019; pp. 102–109. [CrossRef]
60. Osman, M.H.; Kugele, S.; Shafaei, S. Run-Time Safety Monitoring Framework for AI-Based Systems: Automated Driving Cases. *Proc. Asia-Pac. Softw. Eng. Conf. APSEC* **2019**, *2019*, 442–449. [CrossRef]
61. Szenasi, S.; Kertesz, G.; Felde, I.; Nadai, L. Statistical accident analysis supporting the control of autonomous vehicles. *J. Comput. Methods Sci. Eng.* **2021**, *21*, 85–97. [CrossRef]
62. Hortel, J.-B.; Ledent, P.; Marsso, L.; Laugier, C.; Mateescu, R.; Paigwar, A.; Renzaglia, A.; Serwe, W. Verifying Collision Risk Estimation using Autonomous Driving Scenarios Derived from a Formal Model. *J. Intell. Robot. Syst.* **2023**, *107*, 59. [CrossRef]

63. Wang, D.; Fu, W.; Song, Q.; Zhou, J. Potential risk assessment for safe driving of autonomous vehicles, under occluded vision. *Sci. Rep.* **2022**, *12*, 4891. [CrossRef]
64. Song, Y.; Huh, K. Driving and steering collision avoidance system of autonomous vehicle with model predictive control based on non-convex optimization. *Adv. Mech. Eng.* **2021**, *13*, 16878140211027669. [CrossRef]
65. Dosovitskiy, A.; Fischer, P.; Ilg, E.; Hausser, P.; Hazirbas, C.; Golkov, V.; Van Der Smagt, P.; Cremers, D.; Brox, T. FlowNet: Learning optical flow with convolutional networks. *IEEE Int. Conf. Comput. Vis. (ICCV)* **2015**, *2015*, 2758–2766.
66. Mayer, N.; Ilg, E.; Hausser, P.; Fischer, P.; Cremers, D.; Dosovitskiy, A.; Brox, T. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
67. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for autonomous driving? The KITTI vision benchmark suite. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3354–3361.
68. Persistence of Vision Pty, Ltd. Persistence of Vision Raytracer [Computer Software]. 2004. Available online: http://www.povray.org/download/ (accessed on 20 December 2023).
69. Shah, S.; Dey, D.; Lovett, C.; Kapoor, A. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In *Field and Service Robotics*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 621–635.
70. Available online: https://github.com/fanweng/Udacity-Sensor-Fusion-Nanodegree (accessed on 20 December 2023).
71. CARLA. CAR Learning to Act. Available online: https://carla.org/ (accessed on 20 December 2023).
72. Wen, L.; Du, D.; Cai, Z.; Lei, Z.; Chang, M.-C.; Qi, H.; Lim, J.; Yang, M.-H.; Lyu, S. UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking. *Comput. Vis. Image Understand.* **2020**, *193*, 102907. [CrossRef]
73. Real-World Use of Automated Driving Systems and their Safety Consequences: A Naturalistic Driving Data Analysis [Supporting Datasets]. 2020. Available online: https://vtechworks.lib.vt.edu/items/22442930-c5be-40c4-af7d-4c2f1ea8d416 (accessed on 20 December 2023).
74. Houston, J.; Zuidhof, G.; Bergamini, L.; Ye, Y.; Jain, A.; Omari, S.; Iglovikov, V.; Ondruska, P. One thousand and one hours: Self-driving motion prediction dataset. *arXiv* **2020**, arXiv:2006.14480.
75. Rohmer, M.F.E.; Singh, S.P.N. V-rep: A versatile and scalable robot simulation framework. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013.
76. Tian, R.; Li, L.; Yang, K.; Chien, S.; Chen, Y.; Sherony, R. Estimation of the vehicle-pedestrian encounter/conflict risk on the road based on TASI 110-car naturalistic driving data collection. *IEEE Intell. Veh. Symp.* **2014**, *2014*, 623–629.
77. Alexiadis, V.; Colyar, J.; Halkias, J.; Hranac, R.; McHale, G. The next generation simulation program. Institute of Transportation Engineers. *ITE J.* **2004**, *74*, 22.
78. Google. Autonomous Vehicles Annual Disengagement Report. In *California Department of Autonomous Vehicles*; Department of Motor Vehicles (DMV): Sacramento, CA, USA, 2016.
79. Delphi. Autonomous Vehicles Annual Disengagement Report. In *California Department of Autonomous Vehicles*; Department of Motor Vehicles (DMV): Sacramento, CA, USA, 2016.
80. Nissan. Autonomous Vehicles Annual Disengagement Report. In *California Department of Autonomous Vehicles*; Department of Motor Vehicles (DMV): Sacramento, CA, USA, 2016.
81. Mercedes-Benz. Autonomous Vehicles Annual Disengagement Report. In *California Department of Autonomous Vehicles*; Department of Motor Vehicles (DMV): Sacramento, CA, USA, 2016.
82. Volkswagen. Autonomous Vehicles Annual Disengagement Report. In *California Department of Autonomous Vehicles*; Department of Motor Vehicles (DMV): Sacramento, CA, USA, 2016.
83. Virginia Traffic Crash Facts 2014. *Virginia Highway Safety Office, Virginia Department of Motor Vehicles*; Department of Motor Vehicles (DMV): Richmond, VA, USA, 2015.
84. Summary of Motor Vehicle Crashes: 2014 Statewide Statistical Summary. In *New York State Department of Motor Vehicles*; Department of Motor Vehicles (DMV): New York City, NY, USA, 2015.
85. Dezfuli, H.; Benjamin, C.A.; Everett, G.; Maggio, M.; Stamatelatos, R. *NASA Risk Management Handbook*; Publication NASA/SP-2011-3422; NASA: Greenbelt, Maryland, 2011.
86. DADA-2000: Can Driving Accidents be Predicted by Driver Attention? Analyzed by A Benchmark. Available online: https://arxiv.org/abs/1904.12634 (accessed on 20 December 2023).
87. Krajewski, R.; Bock, J.; Kloeker, L.; Eckstein, L. The highD dataset: A drone dataset of naturalistic vehicle trajectories on German highways for validation of highly autonomous driving systems. In Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, USA, 4–7 November 2018; pp. 2118–2125.
88. Theofilatos, A. An Advanced Multi-Faceted Statistical Analysis of Accident Probability and Severity Exploiting High Resolution Traffic and Weather. Ph.D. Thesis, National Technical University of Athens, Athens, Greece, 2015.
89. Fagnant, D.J.; Kockelman, K. Preparing a nation for autonomous vehicles: Opportunities, barriers and policy recommendations. *Transp. Res. PartA.* **2015**, *77*, 167–181. [CrossRef]
90. Clark, F.; Zhang, M. Caltrans PEMS highway sensor average flows by occupancy [Dataset]. *Dryad* **2018**. [CrossRef]
91. *BS ISO 3888-2*; British Standard. Passenger Cars-Test Track for a Severe Lane Change Maneuver-Part 2: Obstacle Avoidance. ISO: Geneva, Switzerland, 2002.
92. Schabenberger, R. ADTF: Framework for driver assistance and safety systems. *VDI BERICHTE* **2007**, *2007*, 701–710.

93. Li, D.; Shi, X.; Long, Q.; Liu, S.; Yang, W.; Wang, F.; Wei, Q.; Qiao, F. *DXSLAM: A Robust and Efficient Visual SLAM System with Deep Features*; IEEE: Las Vegas, NV, USA, 2020.
94. Rong, G.; Shin, B.H.; Tabatabaee, H.; Lu, Q.; Lemke, S.; Možeiko, M.; Boise, E.; Uhm, G.; Gerow, M.; Mehta, S.; et al. Lgsvl simulator: A high fidelity simulator for autonomous driving. In Proceedings of the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 1–6.
95. dSpace. Over-the-Air Simulation of Echoes for Automotive Radar Sensors. Available online: https://www.dspace.com/en/ltd/home/news/engineersinsights/over-the-air-simulation.cfm (accessed on 20 December 2023).
96. Mechanical Simulation Corporation. Unreal Engine Marketplace Showcase. Available online: https://www.carsim.com/publications/newsletter/2021_03_17.php (accessed on 20 December 2023).
97. TASS International. Prescan Overview. Available online: https://tass.plm.automation.siemens.com/prescan-overview (accessed on 20 December 2023).
98. PTV Group. Virtual Testing of Autonomous Vehicles with PTV Vissim. Available online: https://www.ptvgroup.com/en/solutions/products/ptv-vissim/areas-of-application/autonomous-vehicles-and-newmobility/ (accessed on 20 December 2023).
99. Carballo, A.; Lambert, J.; Monrroy, A.; Wong, D.; Narksri, P.; Kitsukawa, Y.; Takeuchi, E.; Kato, S.; Takeda, K. *In Proceedings of the LIBRE: The Multiple 3d Lidar Dataset, Intelligent Vehicles Symposium (IV)*; Las Vegas, NV, USA, 19 October–13 November 2020, IEEE: Toulouse, France, 2020; pp. 1094–1101.
100. Sakaridis, C.; Dai, D.; Van Gool, L. Semantic foggy scene understanding with synthetic data. *Int. J. Comput.* **2018**, *126*, 973–992. [CrossRef]
101. Pitropov, M.; Garcia, D.E.; Rebello, J.; Smart, M.; Wang, C.; Czarnecki, K.; Waslander, S. Canadian adverse driving conditions dataset. *Int. J. Robot. Res.* **2021**, *40*, 681–690. [CrossRef]
102. Yu, F.; Chen, H.; Wang, X.; Xian, W.; Chen, Y.; Liu, F.; Madhavan, V.; Darrell, T. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; IEEE/CVF: Toulouse, France, 2020; pp. 2636–2645.
103. Neuhold, G.; Ollmann, T.; RotaBulo, S.; Kontschieder, P. Themapillaryvistas dataset for semantic understanding of street scenes. In *Proceedings of the International Conference on Computer Vision, ICCV*; Venice, Italy, 22–29 October 2017, IEEE: Toulouse, France, 2017; pp. 4990–4999.
104. Braun, M.; Krebs, S.; Flohr, F.; Gavrila, D.M. Eurocity persons: A novel benchmark for person detection in traffic scenes. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *41*, 1844–1861. [CrossRef] [PubMed]
105. Maddern, W.; Pascoe, G.; Linegar, C.; Newman, P. 1 year, 1000 km: The Oxford robot car dataset. *Int. J.Robot.Res.* **2017**, *36*, 3–15. [CrossRef]
106. Caesar, H.; Bankiti, V.; Lang, A.H.; Vora, S.; Liong, V.E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; Beijbom, O. Nuscenes: Amultimodal dataset for autonomous driving. In Proceedings of the Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; IEEE/CVF: Toulouse, France, 2020; pp. 11621–11631.
107. Che, Z.; Li, G.; Li, T.; Jiang, B.; Shi, X.; Zhang, X.; Lu, Y.; Wu, G.; Liu, Y.; Ye, J. *D*2-city: A large-scale dashcam video dataset of diverse traffic scenarios. *arXiv* **2019**, arXiv:1904.01975.
108. Binas, J.; Neil, D.; Liu, S.-C.; Delbruck, T. DDD17: End-to-endDAVISdriving dataset. *arXiv* **2017**, arXiv:1711.01458.
109. Chang, M.-F.; Lambert, J.; Sangkloy, P.; Singh, J.; Bak, S.; Hartnett, A.; Wang, D.; Carr, P.; Lucey, S.; Ramanan, D.; et al. Argoverse: 3d tracking and forecasting with rich maps. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 8748–8757.
110. Sun, P.; Kretzschmar, H.; Dotiwalla, X.; Chouard, A.; Patnaik, V.; Tsui, P.; Guo, J.; Zhou, Y.; Chai, Y.; Caine, B.; et al. Scalability in perception for autonomous driving: Waymo open dataset. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 2446–2454.
111. Pham, Q.-H.; Sevestre, P.; Pahwa, R.S.; Zhan, H.; Pang, C.H.; Chen, Y.; Mustafa, A.; Chandrasekhar, V.; Lin, J. A*3D dataset: Towards autonomous driving in challenging environments. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 2267–2273.
112. Lei, Y.; Emaru, T.; Ravankar, A.A.; Kobayashi, Y.; Wang, S. Semantic Image Segmentation Snow Driving Scenarios. In Proceedings of the 2020 IEEE International Conference on Mechatronics and Automation (ICMA), Beijing, China, 13–16 October 2020; pp. 1094–1100.
113. Huang, X.; Wang, P.; Cheng, X.; Zhou, D.; Geng, Q.; Yang, R. Theapolloscape open dataset for autonomous driving and its application. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *42*, 2702–2719. [CrossRef]
114. Ros, G.; Sellart, L.; Materzynska, J.; Vazquez, D.; Lopez, A.M. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 3234–3243.
115. Richter, S.R.; Hayder, Z.; Koltun, V. Playing for benchmarks. In Proceedings of the International Conferenceon Computer Vision, ICCV, Venice, Italy, 22–29 October 2017; IEEE: Toulouse, France, 2017; pp. 2213–2222.
116. Liu, D.; Cui, Y.; Cao, Z.; Chen, Y. A large-scale simulation dataset: Boost the detection accuracy for special weather conditions. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; pp. 1–8.
117. Sakaridis, C.; Dai, D.; VanGool, L. ACDC: The adverse conditions dataset with correspondences for semantic driving scene understanding. *arXiv* **2021**, arXiv:2104.13395.

118. Carlevaris-Bianco, N.; Ushani, A.K.; Eustice, R.M. University of Michigan North Campus long-term vision and lidar dataset. *Int. J. Robot. Res.* **2016**, *35*, 1023–1035. [CrossRef]
119. Wenzel, P.; Wang, R.; Yang, N.; Cheng, Q.; Khan, Q.; von Stumberg, L.; Zeller, N.; Cremers, D. 4Seasons: Across-season dataset for multi-weather SLAMin autonomous driving. In *Proceedings of the DAGM German Conference on Pattern Recognition GPCR, Deutsche Arbeitsgemeinschaft für Mustererkennung (DAGM)*; Dortmund, Germany, 10–13 September 2021, DAGM: Bonn, Germany, 2021; pp. 404–417.
120. Tung, F.; Chen, J.; Meng, L.; Little, J.J. The raincouver scene parsing Benchmark for self-driving adverse weather and at night. *Robot. Autom. Lett. (RA-L)* **2017**, *2*, 2188–2193. [CrossRef]
121. Zendel, O.; Honauer, K.; Murschitz, M.; Steininger, D.; Dominguez, G.F. Wildash-creating hazard-aware benchmarks. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; ECCV: Prague, Czech, 2018; pp. 402–416.
122. Choi, Y.; Kim, N.; Hwang, S.; Park, K.; Yoon, J.S.; An, K.; Kweon, I.S. KAIST multi-spectral day/night dataset for autonomous and assisted driving. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 934–948. [CrossRef]
123. Bijelic, M.; Gruber, T.; Mannan, F.; Kraus, F.; Ritter, W.; Dietmayer, K.; Heide, F. Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; IEEE: Toulouse, France, 2020; pp. 11682–11692.
124. Geyer, J.; Kassahun, Y.; Mahmudi, M.; Ricou, X.; Durgesh, R.; Chung, A.S.; Hauswald, L.; Pham, V.H.; Mühlegg, M.; Dorn, S.; et al. A2D2:Audi autonomous driving dataset. *arXiv* **2020**, arXiv:2004.06320.
125. Uřičář, M.; Křížek, P.; Sistu, G.; Yogamani, S. Soilingnet: Soiling Detection Automotive Surround-view cameras. In Proceedings of the Intelligent Transportation Systems Conference, ITSC, Auckland, New Zealand, 27–30 October 2019; IEEE: Toulouse, France, 2019; pp. 67–72.
126. Sheeny, M.; De Pellegrin, E.; Mukherjee, S.; Ahrabian, A.; Wang, S.; Wallace, A. RADIATE: A radar dataset for automotive perception in bad weather. In Proceedings of the International Conference on Robotics and Automation, ICRA, Xi'an, China, 30 May–5 June 2021; IEEE: Toulouse, France, 2021; pp. 1–7.
127. Yan, Z.; Sun, L.; Krajník, T.; Ruichek, Y. EU long-term dataset with multiple sensors for autonomous driving. In Proceedings of the International Conference on Intelligent Robots and Systems, IROS, Las Vegas, NV, USA, 24 October 2020–24 January 2021; IEEE/RSJ: Toulouse, France, 2020; pp. 10697–10704.
128. Basterretxea, K.; Martínez, V.; Echanobe, J.; Gutiérrez-Zaballa, J.; DelCampo, I. HSI-drive: A dataset for the research of hyperspectral image processing applied to autonomous driving systems. In Proceedings of the 2021 IEEE Intelligent Vehicles Symposium (IV), Nagoya, Japan, 11–17 July 2021; pp. 866–873.
129. Bos, J.P.; Chopp, D.; Kurup, A.; Spike, N. Autonomy At the end of the earth: An inclement weather autonomous driving dataset. *Auton. Syst. Sens. Process. Secur. Veh. Infrastruct.* **2020**, *11415*, 36–48.
130. Burnett, K.; Yoon, D.J.; Wu, Y.; Li, A.Z.; Zhang, H.; Lu, S.; Qian, J.; Tseng, W.-K.; Lambert, A.; Leung, K.Y.; et al. Boreas: A multi-season autonomous driving dataset. *arXiv* **2020**, arXiv:2203.10168. [CrossRef]
131. Naseri, M.; Shahid, A.; Gordebeke, G.J.; Lemey, S.; Boes, M.; Van De Velde, S.; De Poorter, E. Machine Learning-Based Angle of Arrival Estimation for Ultra-Wide Band Radios. *IEEE Commun. Lett.* **2022**, *26*, 1273–1277. [CrossRef]
132. Margiani, T.; Cortesi, S.; Keller, M.; Vogt, C.; Polonelli, T.; Magno, M. Angle of Arrival and Centimeter Distance Estimation on a Smart UWB Sensor Node. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 9508110. [CrossRef]
133. Yasmin, R.; Petäjäjärvi, J.; Mikhaylov, K.; Pouttu, A. On the integration of LoRaWAN with the 5G test network. In Proceedings of the 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Montreal, QC, Canada, 8–13 October 2017; pp. 1–6. [CrossRef]
134. Navarro-Ortiz, J.; Sendra, S.; Ameigeiras, P.; Lopez-Soler, J.M. Integration of LoRaWAN and 4G/5G for the Industrial Internet of Things. *IEEE Commun. Mag.* **2018**, *56*, 60–67. [CrossRef]
135. Meijers, J.; Michalopoulos, P.; Motepalli, S.; Zhang, G.; Zhang, S.; Veneris, A.; Jacobsen, H.-A. Blockchain for V2X: Applications and Architectures. *IEEE Open J. Veh. Technol.* **2022**, *3*, 193–209. [CrossRef]
136. Zhang, Y.; Carballo, A.; Yang, H.; Takeda, K. Perception and sensing for autonomous vehicles under adverse weather conditions: A survey. *ISPRS J. Photogramm. Remote Sens.* **2023**, *196*, 146–177. [CrossRef]
137. Leitch, S.G.; Ahmed, Q.Z.; Abbas, W.B.; Hafeez, M.; Laziridis, P.I.; Sureephong, P.; Alade, T. On Indoor Localization Using WiFi, BLE, UWB, and IMU Technologies. *Sensors* **2023**, *23*, 8598. [CrossRef]
138. Manzoni, P.; Calafate, C.T.; Cano, J.-C.; Hernández-Orallo, E. Indoor Vehicles Geolocalization Using LoRaWAN. *Future Internet* **2019**, *11*, 124. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.