



Review

Image-Based Obstacle Detection Methods for the Safe Navigation of Unmanned Vehicles: A Review

Samira Badrloo ^{1,2} , Masood Varshosaz ², Saied Pirasteh ^{1,3,*} and Jonathan Li ⁴

¹ GeoAI, Smarter Map and LiDAR Lab, Department of Surveying and Geoinformatics, Faculty of Geosciences and Environmental Engineering, Southwest Jiaotong University, Xipu Campus, Chengdu 611756, China

² Department of Photogrammetry, K.N. Toosi University of Technology, Tehran 19697, Iran

³ Department of Geotechnics and Geomatics, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai 602105, India

⁴ Geospatial Sensing and Data Intelligence Lab, Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON N2L 3G1, Canada

* Correspondence: sapirasteh@swjtu.edu.cn; Tel.: +86-131-381-9193

Abstract: Mobile robots lack a driver or a pilot and, thus, should be able to detect obstacles autonomously. This paper reviews various image-based obstacle detection techniques employed by unmanned vehicles such as Unmanned Surface Vehicles (USVs), Unmanned Aerial Vehicles (UAVs), and Micro Aerial Vehicles (MAVs). More than 110 papers from 23 high-impact computer science journals, which were published over the past 20 years, were reviewed. The techniques were divided into monocular and stereo. The former uses a single camera, while the latter makes use of images taken by two synchronised cameras. Monocular obstacle detection methods are discussed in appearance-based, motion-based, depth-based, and expansion-based categories. Monocular obstacle detection approaches have simple, fast, and straightforward computations. Thus, they are more suited for robots like MAVs and compact UAVs, which usually are small and have limited processing power. On the other hand, stereo-based methods use pair(s) of synchronised cameras to generate a real-time 3D map from the surrounding objects to locate the obstacles. Stereo-based approaches have been classified into Inverse Perspective Mapping (IPM)-based and disparity histogram-based methods. Whether aerial or terrestrial, disparity histogram-based methods suffer from common problems: computational complexity, sensitivity to illumination changes, and the need for accurate camera calibration, especially when implemented on small robots. In addition, until recently, both monocular and stereo methods relied on conventional image processing techniques and, thus, did not meet the requirements of real-time applications. Therefore, deep learning networks have been the centre of focus in recent years to develop fast and reliable obstacle detection solutions. However, we observed that despite significant progress, deep learning techniques also face difficulties in complex and unknown environments where objects of varying types and shapes are present. The review suggests that detecting narrow and small, moving obstacles and fast obstacle detection are the most challenging problem to focus on in future studies.

Keywords: obstacle detection; image-based; UAV; MAVs; deep learning methods



Citation: Badrloo, S.; Varshosaz, M.; Pirasteh, S.; Li, J. Image-Based Obstacle Detection Methods for the Safe Navigation of Unmanned Vehicles: A Review. *Remote Sens.* **2022**, *14*, 3824. <https://doi.org/10.3390/rs14153824>

Academic Editors: Francesco Nex, Davide Antonio Cucci and Eufemia Tarantino

Received: 29 April 2022

Accepted: 4 August 2022

Published: 8 August 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The use of mobile robots such as Unmanned Aerial Vehicles (UAVs), Unmanned Ground Vehicles (UGVs) has increased in recent years for photogrammetry [1–3], and many other applications (see Table 1). A remotely piloted robot should be able to detect obstacles automatically. In general, obstacle detection techniques can be divided into three groups: Image-based [4,5], sensor-based [6,7], and hybrid [8,9]. In sensor-based methods, various active sensors such as lasers [10–13], radar [14,15], sonar [16], ultrasonic [17,18], and Kinect [19] have been used.

Table 1. The recent papers on robot applications.

Title	Ref	Applications
Design of a UGV powered by solar energy for precision agriculture	[20]	Agriculture
A new algorithm for landslide geometric and deformation analysis supported by digital elevation models	[21]	Automated derivation of landslide geometry
Double adaptive intensity-threshold method for uneven lidar data to extract road markings	[22]	Autonomous driving
High-resolution terrain modelling using airborne LiDAR data with transfer learning	[23]	Terrain modelling
Optimising sector ring histogram of oriented gradients for human injured detection from drone images	[24]	Finding injured people
Using geometric constraints to improve the performance of image classifiers for automatic segmentation of traffic signs	[25]	Autonomous driving
Embedded design and implementation of a mobile robot for surveillance applications	[26]	Surveillance
Assessing the impact of understory vegetation density on solid obstacle detection for off-road autonomous ground vehicles	[27]	Autonomous driving
Fast-PGMED: Fast and Dense Elevation Determination for Earthwork Using Drone and Deep Learning	[28]	Elevation determination

Sensor-based methods have their own merits and disadvantages. For instance, in addition to being reasonably priced, sonar and ultrasonic sensors can determine the direction and position of an obstacle. However, sonar and ultrasonic waves are affected by both constructive and destructive interference of ultrasonic reflections from multiple environmental obstacles [29]. In some situations, radar waves may be an excellent alternate, mainly when no visual data is available. Nevertheless, radar sensors are not small or light, which means installing them on small robots is not always feasible [30,31]. Moreover, infrared waves have a limited Field Of View (FOV), and their performance is dependent on weather conditions [32]. Despite being a popular sensor, LiDAR is relatively large and, thus, cannot permanently be installed on small robots like MAVs. Therefore, despite their popularity and ease of use, active sensors may not be ideal for obstacle detection when weight, size, energy consumption, sensitivity to weather conditions, and radio-frequency interference issues matter [32,33].

Alternative to active sensors is lightweight cameras that provide visual information about the environment the robot travels in. Cameras are passive sensors and are used by numerous image-based algorithms to detect obstacles using grayscale values [34] and point [35] or edge [36,37] features. They can provide details regarding the amount of the robot's movement or displacement and the obstacle's colour, shape, and size [38]. In addition to enabling real-time and safe obstacle detection, image-based techniques are not disturbed by environmental electromagnetic noises. Additionally, the visual data can be used to guide the robot through various image-based navigation techniques currently available.

Hybrid techniques integrate data from active and passive sensors, which would benefit and also suffer from the weaknesses of both systems [39–41]. Several studies have been carried out in this field (see Table 2). However, it should be noted that fusing the data from multiple sources has difficulties and complexities, such as accurate calibration of multi-sensor systems [8,42,43]. Additionally, only a few studies have been conducted on the real-time use of multiple heterogeneous sensors, typically asynchronous [42]. The data collected by various sensors is not homogeneous and usually requires complicated data processing.

Table 2. The recent papers on hybrid obstacle detection techniques.

Title	Ref	Content
Multimodal obstacle detection and collision avoidance for micro aerial vehicles	[44]	Developing a lightweight 3D laser scanner and visual obstacle detection using wide-angle stereo cameras
Multilayered mapping and navigation for autonomous micro aerial vehicles	[45]	Integrating a three-dimensional (3D) laser scanner, two stereo camera pairs, and ultrasonic distance sensors
Autonomous walking with a guiding stick for the blind using echolocation and image processing	[46]	Using ultrasonic sensors, image sensors and a Smartphone app to navigate the user to the destination
Obstacle detection system for small UAVs using ads-b and thermal imaging.	[47]	Integrating a TIR camera and an Automatic Dependent SurveillanceBroadcast (ADS-B) receiver
Autonomous navigation in a warehouse with a cognitive micro aerial vehicle	[48]	Integrating a dual 3D laser scanner, three stereo camera pairs, an IMU, an RFID reader, and a powerful onboard computer running the ROS middleware
So-net: joint semantic segmentation and obstacle detection using a deep fusion of monocular camera and radar	[49]	Using radar and vision-based deep learning perception framework
Multimodal obstacle detection in unstructured environments with conditional random fields	[50]	Fusing lidar and camera
Deep feature-level sensor fusion using skip connections for real-time object detection in autonomous driving	[9]	Sensor fusion of the visible camera with the millimetre-wave radar and the thermal camera

This paper reviews image-based obstacle detection techniques that can acquire and analyse visual information from their environment. We discuss the strengths and weaknesses of state-of-the-art image-based obstacle detection methods developed for unmanned vehicles like UAVs and USVs. The remaining part of this article consists of these sections. In Section 2, the presented techniques are classified, and the main criteria used to evaluate them are described. Then, in Sections 3 and 4, image-based obstacle detection methods are reviewed and discussed. Then in Section 5, overall discussions and future prospects are presented mainly based on the authors' perspectives. Finally, conclusions are made in Section 6, where suggestions for future research are also presented.

2. Classification of Obstacle Detection Techniques

Different papers have been published on image-based obstacle detection in the last two decades. The papers used in this review have all been extracted from journals. Various databases were searched with terms like "vision-based obstacle detection", "monocular obstacle detection", "stereo-based obstacle detection", "UAV obstacle detection", "combining active sensors and cameras, hybrid methods", and "sensors and navigation". Hundreds of papers were studied, more than 110 of which have been used here to develop our review framework and shape the presented discussions.

Image-based obstacle detection methods can be divided into monocular and stereo (Figure 1). The former uses a single image, whereas the latter utilises images captured by two synchronised cameras. Monocular methods are discussed in four groups: appearance-based, motion-based, depth-based, and expansion-based. Stereo-based approaches are studied separately in IPM-based and disparity histogram-based categories.

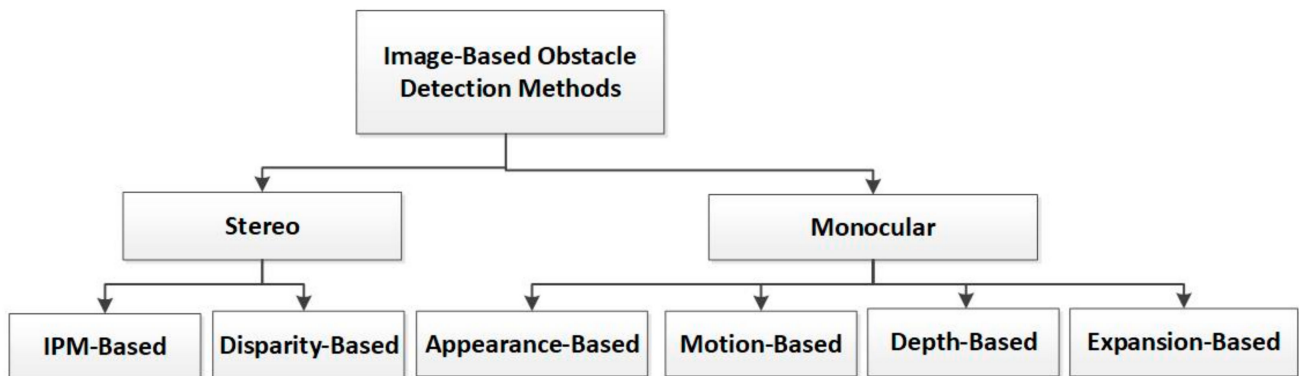


Figure 1. Classification of image-based obstacle detection methods.

To ensure safe and accurate navigation, an obstacle detection algorithm should have the following abilities:

- (a) Narrow and small obstacle detection

Narrow and small obstacles, such as rope, wire, and tree branches, are represented by a few pixels in an image. Thus, they are challenging to detect, and an accident can easily happen if they are not adequately identified.

- (b) Moving obstacle detection

The robot must recognise moving obstacles such as birds, mobile robots, and humans.

- (c) Obstacle detection in all directions

In complex environments, the detection should not only be carried out in the moving direction but also in all other directions; otherwise, objects (e.g., a bird) approaching the robot from an angle not included in its FOV can disturb its operation or even cause it to crash.

- (d) Fast/real-time obstacle detection

The response speed of a robot to obstacles depends on the speed of the obstacle detection algorithm. It can crash if the algorithm is not fast enough (relative to the robot's speed).

For the sake of simplicity, the terms Narrow and Small Obstacle Detection (NSOD), Moving Obstacle Detection (MOD), Obstacle Detection in All Directions (ODAD), and Fast Obstacle Detection (FOD) are used in this paper. In the following, monocular and stereo image-based obstacle detection techniques are discussed in detail. At the end of each section or subsection, papers published since 2015 are also listed to indicate current trends and interests of researchers. The last four columns of each Table show whether any of the above criteria are addressed in a paper.

3. Monocular Obstacle Detection Techniques

Monocular image-based obstacle detection methods use a single camera mounted in front of or around the robot (Figure 2). They are divided into four categories [51]: appearance-based [32,52], motion-based [53], depth-based [54], and expansion-based [5,35,55]. Details will follow.

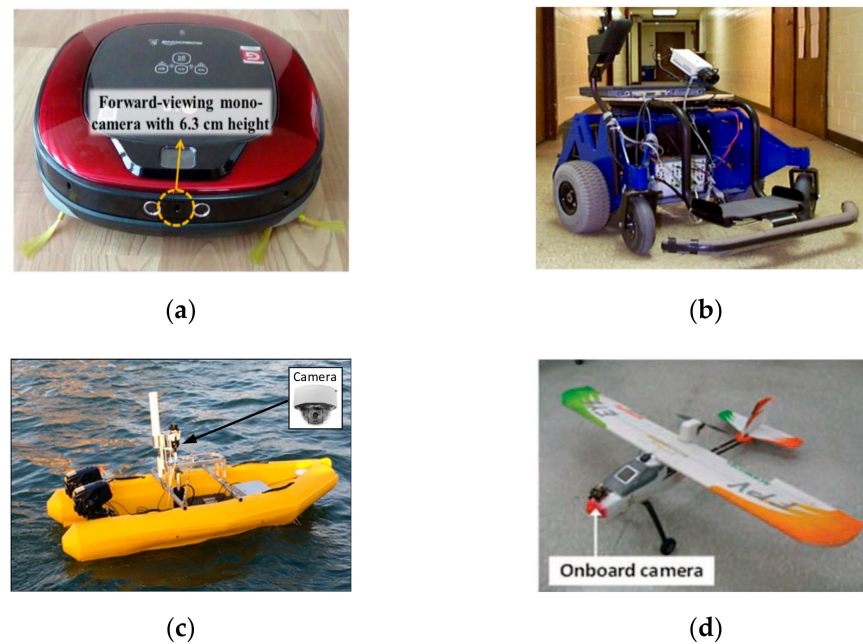


Figure 2. A single camera mounted in front of the mobile robot: (a) [32]; (b) [56]; (c) [57]; (d) [30].

3.1. Appearance-Based

These methods consider an obstacle as a foreground object against a uniform background (i.e., ground or sky). They work based on some prior knowledge from the relevant background in the form of the edge [30], colour [34], texture [56], or shape [32] features. Obstacle detection is performed on single images taken sequentially using a camera mounted in front of the robot. The acquired image is examined to see if it conforms to that of sky or ground features; if it does not, it is considered an obstacle pixel. This process is performed for every pixel in the image. The result is a binary image in which obstacles are presented in white and the rest in black pixels.

In terrestrial robots, the ground data such as road or floor are detected first. Then, to detect the obstacles, ground data is used. Ulrich and Nourbakhsh [56] proposed a technique where each pixel in the image is labelled as an obstacle or ground using its pixel values. Their system is trained by moving the robot within different environments. As a result, in practical situations, if the illumination conditions vary from those used during the training phase, obstacles will not effectively be identified. In another study, Lee et al. [32] used Markov's Random Field (MRF) segmentation to detect small and narrow obstacles in an indoor environment. However, the camera should not be more than 6.3 cm away from the ground [32]. An omnidirectional camera system and the Hue Saturation Value (HSV) colour model were used to separate obstacles by Shih An et al. [58]. They created a binary image and filtered out the noise using a width-first search technique to cope with image noise. In another study [59], object-based background subtraction and image-based obstacle detection techniques were used for static and moving objects. They used a single wide-angle camera for real-time obstacle detection. Moreover, Liu et al. [57] proposed a real-time monocular obstacle detection method to identify the water horizon line and saliency estimation for USVs like boats or ships. The system was developed to detect objects below the water edge that may pose a threat to USVs. They claimed their method outperforms similar state-of-the-art techniques [57].

Conventional image processing techniques do not usually meet the expectations of real-time applications. Therefore, recent research has focused on increasing the speed of obstacle detection using Convolutional Neural Networks (CNNs). For example, to increase the speed, Talele et al. [60] used TensorFlow [61] and OpenCV [62] to detect obstacles by scanning the ground for distinct pixels and classifying them as obstacles. Similarly, Rane et al. [63] used TensorFlow to identify pixels different from the ground. Their method

was real-time and applicable to various environments. To recognise and track typical moving obstacles, Qiu et al. [52] used YOLOv3 and Simple Online and Real-time Tracking (SORT). To solve the low accuracy and slow reaction time of existing detection systems, the You Only Look Once v4 (YOLOv4) network, which is an upgraded version of YOLO [64], was proposed by He et al. [65]. It improved the recognition of obstacles at medium and long distances [65].

Furthermore, He and Liu [66] developed a real-time technique for fusing features to boost the effectiveness of detecting obstacles in misty conditions. Additionally, Liu et al. [67] introduced a novel semantic segmentation algorithm based on a spatially constrained mixture model for real-time obstacle detection in marine environments. A Prior Estimation Network (PEN) was proposed to improve the mixture model.

As for airborne robots, most research looks for a way to separate the sky from the ground. For example, Huh et al. [30] separated the sky from the ground using a horizon line. They then determined moving obstacles using the particle filter algorithm. Their method could be used in complex environments and low-altitude flights. Despite being efficient in detecting moving obstacles, their technique could not be used for stationary obstacles [30]. In another study, a method was introduced by Mashaly et al. [34] to find the sky in a complex environment, with obstacles separated from the sky in a binary image [34]. De Croon and De Wagter [68] suggested a self-supervised learning method to discover the horizon line [68]. In another study [59], a single wide-angle camera was used for real-time obstacle detection. The object-based background subtraction and image-based obstacle detection techniques were used for static and moving objects.

The strengths and weaknesses of many appearance-based algorithms published between 2015 and 2021 are summarised in Table 3. As can be seen, appearance-based methods have been used on various robots. Every study has had its concerns. One research has investigated narrow and small obstacle detection (i.e., [32]), whereas moving obstacle detection has been the centre of focus in a few others. Except for Shih An et al. [58], other articles have focused on detecting obstacles in front of the robot. It is also evident that the attention in the last three years has been on the speed of obstacle detection procedures. Appearance-based methods are generally limited to environments where obstacles can easily be distinguished from the background. This assumption can easily be violated, particularly in complex environments containing objects, like buildings, trees, and humans [69] with varying shapes and colours. Some of the techniques are affected by the distance to the object or the noise in the images. Moreover, when using deep learning approaches, the detection of obstacles is primarily affected by the change in the environment and the number and types of samples in the training data set. Therefore, we suggest using a semantic segmentation algorithm performed by deep learning networks, provided that sufficient training data is available. Another alternative would be to enrich appearance-based methods by providing them with distance to object data that can be obtained using a sensor or a depth-based algorithm (see the following sections).

Table 3. The recent papers on appearance-based obstacle detection techniques.

Ref	Used for:	Principle	Strength(s)	Weakness(es)	NSOD	MOD	ODAD	FOD
[30]	UAV	Detection and tracking airborne obstacles in a complex background	Moving obstacle detection	Inability to detect stationary obstacles	-	✓	-	-
[34]	UAV	Sky segmentation	Efficiency in complex environments	Not separating far and near obstacles	-	-	-	-

Table 3. Cont.

Ref	Used for:	Principle	Strength(s)	Weakness(es)	NSOD	MOD	ODAD	FOD
[32]	Car	Inverse perspective mapping is used to obtain geometric cues for obstacle detection	Small and narrow obstacle detection	Camera installation at a little distance from the ground. Inefficiency for UAVs and cars	✓	-	-	-
[68]	Mobile robot	Self-supervised learning to find the horizon line	Efficiency in different environments	Unsuccessful in office environments	-	-	-	-
[58]	Mobile robot	Combining an improved dynamic window approach (IDWA) and an artificial potential field to avoid obstacles	- Omnidirectional obstacle detection - Noise-reducing, which makes obstacle detection more precise	Insufficient in real-world environments	-	-	✓	-
[60]	Mobile robot	Use of TensorFlow and OpenCV	Real-time	The resolution of the camera is very poor, which affects the accuracy of obstacle detection	-	-	-	✓
[63]	Mobile robot	Use of TensorFlow	Real-time	Low-resolution images	-	-	-	✓
[52]	Agriculture machine	Use of improved YOLOv3 and Deep SORT	Moving obstacle detection Real-time	Requiring enriched training data is	-	✓	-	✓
[59]	Dual-lens camera sensor robot	Use of a local object background subtraction method	- Use of a wide-angle camera Real-time	Incomplete obstacle detection	-	✓	-	✓
[65]	Train	Use of the improved-YOLOv4 network	High accuracy and real-time obstacle detection	The environment, the amount and type of data impact	-	-	-	✓
[57]	USV	Horizon line detection to detect obstacles below the estimated horizon line Feature fusion to improve the performance of the camera sensor in obstacle detection under misty weather	Real-time	Inefficiency for cars and UAVs	-	-	-	✓
[66]	Car	Semantic segmentation for real-time obstacle detection	Obstacle detection under foggy weather Real-time	Low accuracy	-	-	-	✓
[67]	USV	Semantic segmentation for real-time obstacle detection	Real-time	Weak navigation when there is a strong reflection and confusing obstacles	-	-	-	✓

3.2. Motion-Based

In motion-based methods, it is assumed that nearby objects have sharp movements that can be detected using motion vectors in the image. The process involves taking two successive images or frames in a very short time. At first, several match points are extracted on both frames. Then, the displacement vectors of the match points are computed. Since objects closer to the camera have larger displacements, any point with a displacement value that exceeds a particular threshold is considered an obstacle pixel.

Various studies have been conducted in this field. Jia et al. [70] introduced a novel method that uses motion features to distinguish obstacles from shadows and road markings. Instead of using all pixels, they only used corners and Scale Invariant Feature Transform (SIFT) features to achieve real-time obstacle detection. Such an algorithm can fail if the number of mismatched features is high [70].

Optical flow is the data used in most motion-based approaches. Ohnishi and Imiya [71] prevented a mobile robot from colliding with obstacles without having a map of the environment. Gharani and Karimi [53] used two consecutive frames to estimate the optical flow for obstacle detection on smartphones to help visually impaired people navigate indoor environments. Using a context-aware combination data method, they determined the distance between two consecutive frames. Tsai et al. [72] used Support Vector Machine (SVM) [73] to validate Speeded-up Robust Features (SURF) [74] point detector locations as obstacles. In this research dense optical flow approach was used to extract the data for training SVM. Then, they used obstacle points and measures related to the spatial weighted saliency map to find the obstacle locations. The algorithm presented in their research applies to mobile robots with a camera installed at low altitudes. Consequently, it might not be possible to use it on UAVs that usually fly at high altitudes.

Table 4 summarises the articles published since 2015 on motion-based methods. Except for research by Jia et al. [70], none of the four mentioned criteria were investigated in these studies.

Table 4. Recent papers on motion-based obstacle detection techniques.

Ref	Used for:	Principle	Strength(s)	Weakness(es)	NSOD	MOD	ODAD	FOD
[70]	Car	Use of motion features	Real-time performance by utilising feature points rather than all pixels	Algorithm failure occurs if there are too many miss detections or mismatched feature points	-	-	-	✓
[53]	Blind people	Computing optical flow and tracking certain points	Both moving and stationary obstacle detection	Incorrect detection of some points on lamps, floors, and reflective surfaces	-	-	-	-
[72]	Mobile robot	Combining obstacle points and salience map	Effective obstacle detection in outdoor environments	May not be used on UAVs that usually fly at high altitudes	-	-	-	-

Motion-based obstacle detection relies mainly on the quality of the matching points. Thus, its quality can decrease if the number of mismatched features is high. In addition, if the optical flow is used for motion estimating, care must be taken for image points close to the centre. This is because, in optical flow, the number of motion vectors is not high. Indeed, detecting obstacles in front of the robot using optical flow is still challenging [75,76]. To resolve this problem, we suggest using an expansion-based approach to detect obstacles in the central parts of the images. This integration ensures the strength of the expansion-based technique in detecting frontal objects is employed, while the other parts are analysed by the motion-based method. Alternatively, we may use deep learning networks to solve such problems.

3.3. Depth-Based

Like motion-based methods, depth-based approaches obtain depth information from images taken by a single camera. There are two ways to accomplish this, the first being motion stereo and the second being deep learning. Two cameras are placed on the robot's sides in the former, and a pair of consecutive images are captured. Although these images are only taken using a single camera, they can be considered as a pair of stereo images,

from which the depth of object points can be estimated. For this, the images are searched for matching points. Then, using standard depth estimation calculations [77], the depth of object points is computed. Pixels whose depth is less than a threshold value are regarded as obstacles.

A recent alternative to the above process is employing a deep learning network. At first, the network is trained using appropriate data, so it can produce a depth map from a single image [78]. Samples of such networks can be found in [79,80]. The process then tests any image taken by the robot's camera to determine the depth of its pixels. Then, similar to a classic approach, pixels having a depth smaller than a threshold are considered obstacles.

However, instead of using motion vectors, a complete three-dimensional model of the surroundings is constructed and used to detect nearby obstacles [81,82]. Some of these methods use motion stereo. For example, Häne et al. [83] used motion stereo to produce depth maps using four fisheye images (Figure 3). In this system, an object on the ground was considered an obstacle. Such an algorithm cannot detect moving obstacles. Moreover, it provides a complete map of the environment which requires complex computations.

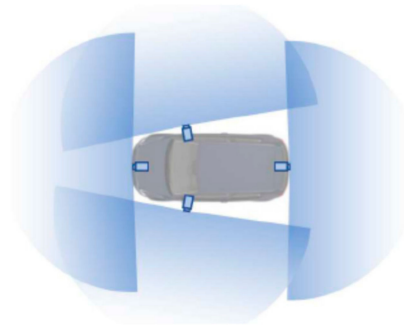


Figure 3. Fisheye cameras mounted on the front, rear and side of the car [83].

In another research, Lin et al. [84] used a fisheye camera and an Inertial Measurement Unit (IMU) for autonomous navigation of a MAV. As part of the system, an algorithm was developed to detect obstacles in a wide FOV using fisheye images. Each fisheye image was converted into two pinhole images without distortion with a sum horizontal viewing angle of 180° . Depth estimation was based on keyframes. Because the depth can only be estimated when the drone moves, this system will not work on MAVs when in hovering mode. Moreover, as the quality of the parts on the sides of a fisheye image is low, the accuracy of the resulting depth image can be low. Besides, the production of two horizontal pinhole images can decrease the vertical FOV and, thus, limit the areas where the obstacles can be detected.

Artificial neural networks and deep learning have been used to estimate depth in recent years [85]. Contrary to methods like 3D model construction, deep learning-based techniques do not require complex computations for obstacle detection. Kumar et al. [86] used four single fisheye images and a CNN to estimate the depth in all directions. They used LiDAR data as ground truth for depth estimation to train the network. The dataset they used in their self-driving car had a 64-beam Velodyne LiDAR and four wide-angle fisheye cameras. In this study, the distortion of the fisheye image was not corrected. It is recommended to improve the results by using more consecutive frames to exploit the motion parallax and better CNN encoders [86]. Their research required further training. Therefore, another future goal for this work is to improve semi-supervised learning using synthetic data and run unsupervised learning algorithms [86]. In another study, Mancini et al. [87] developed a new CNN framework that uses image features obtained via fine-tuning the VGG19 network to compute the depth and consequently detect obstacles.

Moreover, Haseeb et al. [88] presented DisNet, a distance estimation system based on multi-hidden-layer neural networks. They evaluated the system under static conditions, while evaluation of the system mounted on a moving locomotive remained a challenge. In

another research, Hatch et al. [54] presented an obstacle avoidance system for small UAVs, in which the depth is computed using a vision algorithm. The system works by incorporating a high-level control network, a collision prediction network, and a contingency policy. Urban and Caplier [76] developed a navigation module for visually impaired pedestrians, using a video camera in an intelligent light-weighted glasses device. It includes two modules: a static data extractor and a dynamic data extractor. The first is a convolutional neural network used to determine the obstacle's location and distance from the robot. In contrast, using a fully connected neural network, the dynamic data extractor computes the Time-to-Collision by stacking the obstacle data from multiple frames.

Furthermore, some researchers have developed methods to create and use a semantic map of the environment to recognise obstacles [89,90]. A semantic map is a representation of the robot's environment that incorporates both geometric (e.g., height, roughness) and semantic data (e.g., navigation-relevant classes such as trail, grass, obstacle, etc.) [91]. When used in urban autonomous vehicle applications, they can provide autonomous vehicles with a longer sensing range and more excellent manoeuvrability than onboard sensory devices. Some studies have used multi-sensor fusion to improve the robustness of their segmentation algorithms to create semantic maps. More details can be found in [92–96].

Table 5 presents several papers that use depth-based approaches for obstacle detection. As can be seen, the main concern is detecting obstacles on keyframes or using deep learning approaches to improve the speed. In addition, the detection of objects in all directions has been looked at, as the development of autonomous cars and intelligent navigation systems for MAVs have become hot topics in recent years.

Table 5. Recent papers on depth-based obstacle detection techniques.

Ref	Used for:	Principle	Strength(s)	Weakness(es)	NSOD	MOD	ODAD	FOD
[83]	Car	Depth map generation with motion stereo system	Obstacle detection in all directions	Inability to identify moving obstacles	-	-	✓	-
[84]	MAV	Depth estimation based on keyframes	Obstacle detection with wide FOV	- MAV is unable to detect impediments when hovering - Low-quality fish-eye images have a negative impact on detection accuracy	-	-	✓	-
[86]	Car	Use of CNN network	Obstacle detection in all directions	Need for appropriate training data	-	-	✓	-
[87]	UAV	Use of image features obtained via fine-tuning the VGG19 network	Fast obstacle detection	The need for adequate initial training	-	-	-	✓
[88]	-	Use of Multi Hidden-Layer Neural Network, named DisNet	Reliable estimation of distances in static railway scenes	Extraction of inaccurate object bounding boxes that limit obstacle detection	-	-	-	-
[54]	UAV	Use of hybrid neural network	Efficiency in a complex outdoor environment at comparatively high speeds	Poor performance due to noise	-	-	-	-
[76]	visually impaired	Focus on the Time-to-Collision network's ability	Time-efficient	Inability to correctly detect different types of obstacles	-	-	-	✓

In summary, depth-based approaches are well suited for detecting stationary obstacles. Most traditional techniques require a complete map of the environment, which needs intensive computations. Although deep learning-based depth estimation has widely been employed in recent years, some issues still need attention. For example, to achieve good results, the network must be deep, as we know. This, in turn, increases memory usage and calculation complexity [97]. Another critical issue is that extensive high-quality training data is usually needed to train the network properly, which may not readily be available, especially for complex environments. Indeed, for a system to be practical, calculating the depth in complex scenes remains challenging. In situations where the objects to be identified are not diverse, a solution would be to use a semantic segmentation approach to identify the object's pixels in the images. After that, a depth estimation algorithm can be used to estimate the distance between the objects and the camera mounted on the robot.

3.4. Expansion-Based

These methods employ the same principle used by humans to detect obstacles, i.e., the object expansion rate between consecutive images. As we know, an object continuously grows larger when it approaches. Thus determining obstacles, points and/or regions on two sequential images can be used to estimate the object's enlargement value. This value could be computed between homologous areas, distances, or even the SIFT scales of the extracted points. In expansion-based algorithms, if the enlargement value relating to an object exceeds a specific threshold, that object is considered an obstacle.

Expansion-based methods use the objects' enlargement rate in between successive images. They use a concept similar to human perception. Several expansion-based studies have been conducted. In these methods, the obstacle is defined as an object enlarged or resized in consecutive frames. Therefore, sequential frames and various enlargement criteria are used to detect obstacles. For example, Mori and Scherer [98] used the characteristics of the SURF algorithm to detect the initial positions of obstacles that differed in size. This algorithm has simple calculations but may fail due to the slow reaction time to obstacles. Zeng et al. [69] used edge motion in two successive frames to identify approaching obstacles in another research. If the object's edge shifts outwards (relative to its centre in successive frames), the object becomes large [69]. This approach applies to both fixed and mobile robots when the background is homogeneous. However, if the background is complicated, this approach only applies to static objects. Aguilar et al. [99] only detected obstacles conforming to some primary patterns. They use this concept to detect specific obstacles. As a result, obstacles other than those following the predefined patterns cannot be identified.

To detect obstacles, Al-Kaff et al. [35] used SIFT [100] to extract and match points across successive frames. He then formed the convex hull of the matched points. The points were regarded as obstacle points if the change in their SIFT scale values and the convex hull area exceeded a certain threshold (Figure 4). The technique may simultaneously identify both near and far points as obstacles. As a result, the mobile robot will have limited manoeuvrability in complex environments. In addition, the ratio of change of the convex hull region criterion will lose its efficiency if the corresponding points are wrong.

Badrloo and Varshosaz [55] used points with an average distance ratio greater than a specified threshold to identify obstacle points to solve this problem. Their technique was able to distinguish far and near obstacles properly. Others have solved the problem in different ways. For example, like Badrloo and Varshosaz [55], Euclidean distance was acquired between each point and the centroid of all other matched points by Padhy et al. [37]. Escobar et al. [101] computed the optical flow to obtain the expansion rate for obstacle recognition in unknown and complex environments in another study. In another study, Badrloo et al. [102] used the expansion rate of region areas for accurate obstacle detection.

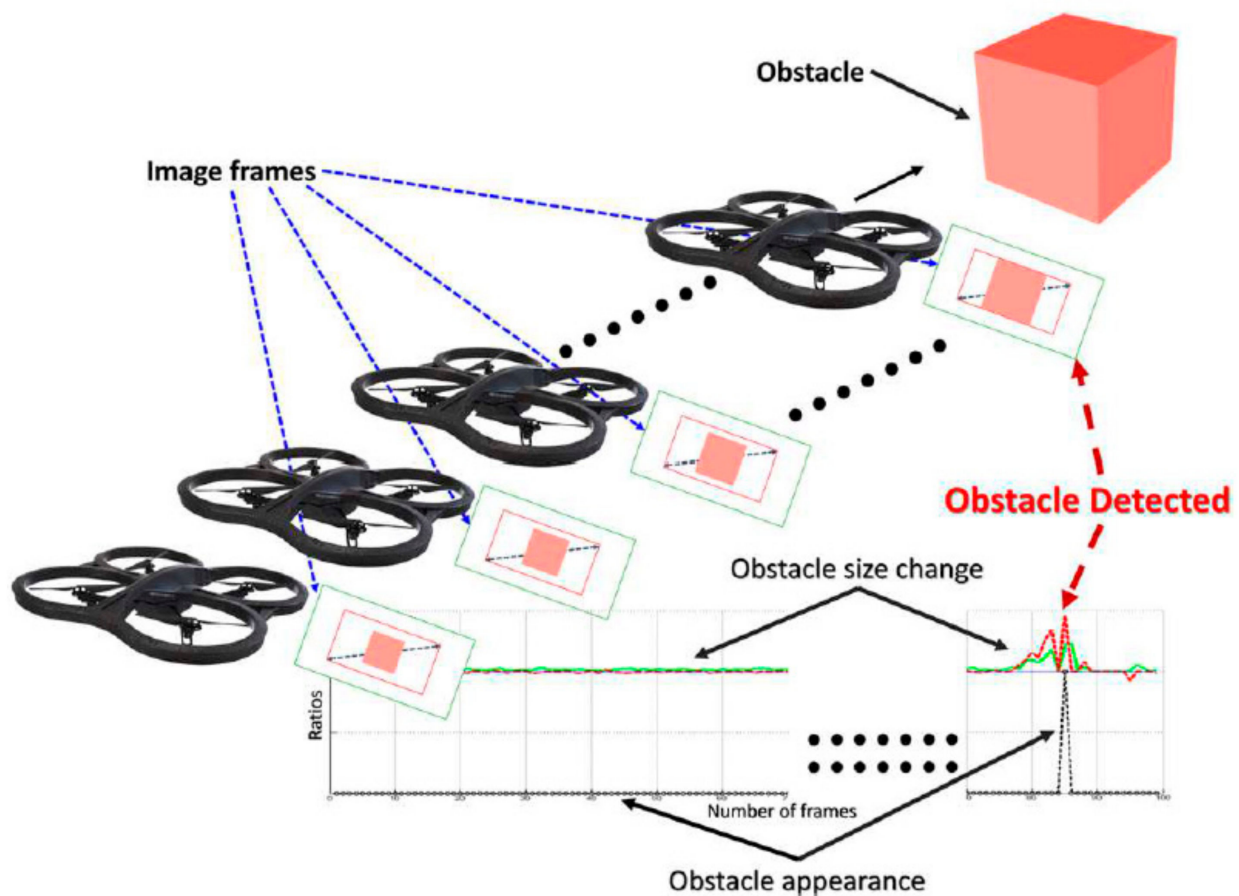


Figure 4. The change in obstacle size is demonstrated by approaching it [35].

Recently, deep learning solutions have been proposed to improve both the speed and the accuracy of obstacle detection, especially in complex and unknown environments. For instance, Lee et al. [5] detected obstacle trees in tree plantations. They trained a machine learning model, the so-called Faster Region-based Convolutional Neural Network (Faster R-CNN), to detect tree trunks for drone navigation. This approach uses the ratio of an obstacle height in the image to the image height. Additionally, the image widths between trees were used to find obstacle-free pathways.

Compared to other monocular techniques, expansion-based approaches employ a simple principle, i.e., the expansion rate. Such techniques are fast, as they do not require extensive computations. However, they may fail when the surrounding objects become complex. Thus, in recent years, deep neural networks have been employed to meet the expectations of real-time applications.

Table 6 summarises the papers reviewed in this section. As can be seen, most techniques have been implemented for flying robots (i.e., MAV and UAV). Only one paper has addressed moving object detection, and only one has aimed for fast detection of obstacles. Perhaps, this has been because, by default, expansion-based techniques are fast. Thus, detection accuracy in complex environments seems to be the primary concern, with little success.

Table 6. Recent papers on expansion-based obstacle detection techniques.

Ref	Used for:	Principle	Strength(s)	Weakness(es)	NSOD	MOD	ODAD	FOD
[69]	UAV	Brain-inspired rasterisation	Efficiency for both fixed and mobile cameras in a simple background	In the complex background, this approach does not apply to mobile robots	-	✓	-	-
[99]	compact MAV	Use of SURF	Reduced computational complexity	Weak in the identification of other obstacles such as trees	-	-	-	-
[35]	MAV	Use of keypoints scale ratio and convex hull area ratio	Can be used in complex environments	- Lack of distinguishing between near-far obstacles - The impact of corresponding incorrect points	-	-	-	-
[55]	UAV	Use of distance ratio	Ability to distinguish near-far obstacles	Moving the camera loses its effectiveness	-	-	-	-
[101]	MAV	Use of optical flow	Navigation in complex environments	Problems with the use of optical flow	-	-	-	-
[37]	UAV	Use of Euclidean distance expansion	Depth estimation from a single camera	- Inefficiency when the UAV's velocity increases - Inefficiency when the obstacle does not have proper texture	-	-	-	-
[5]	UAV	Use of Faster R-CNN and image heights of trees	Ability to detect trees	Unable to detect objects other than trees (e.g., people, cars)	-	-	-	✓
[102]	MAV	Use of regions expansion rate	Accurate and complete obstacle detection	Time-consuming	-	-	✓	-

As seen from the above, expansion-based approaches use points or convex hulls to detect obstacles to increase speed. This leads to the inclusion of incomplete obstacle shapes, which can limit its accuracy. A recent method provides regions of an obstacle for complete and precise obstacle detection [102], although it does not yet meet the requirements of real-time applications. We suggest using methods based on deep neural networks to accelerate the complete and precise detection of obstacles in expansion-based methods.

3.5. Summary

In this section, monocular obstacle detection techniques were reviewed. Due to their high speed and low computational complexity, most obstacle detection techniques today are monocular. Yet they need to advance from different points of view. Appearance-based methods are generally limited to environments where obstacles can easily be distinguished from the background. This assumption can easily be violated, particularly in complex environments containing objects of varying shapes and colours, such as buildings, trees, and humans [69]. Moreover, when using deep learning approaches, the detection of obstacles is primarily affected by the change in the environment and the number and types of samples in the training data set. We suggested using or combining them with a distance to object calculation method. From the review, we saw that the main attention in recent years had been the speed of obstacle detection.

Motion-based obstacle detection relies mainly on the quality of the matching points. Thus, its quality can decrease if the number of mismatched features is high. Indeed, detecting obstacles in front of the robot using optical flow is still a challenge [75,76]. The

current trend is to use deep learning networks to solve such problems. Therefore, we suggest using another method, such as the expansion-based, for the centre part of the image and a motion-based method for the rest.

As for the depth-based methods, they are well suited for detecting stationary obstacles. Although producing a complete depth map of the surroundings is not the goal of obstacle detection, most depth-based algorithms need such data to work, which is computationally intensive. Therefore, we propose to only extract objects in 2D and then use a depth-based approach only to compute the depth of the identified objects. Several papers also observed that in using depth-based methods, the main trend is to detect obstacles on keyframes or to use deep learning approaches to improve the speed. In addition, the detection of objects in all directions has been looked at, as the development of autonomous cars and intelligent navigation systems for MAVs have become hot topics in recent years.

Finally, compared to the other monocular methods, expansion-based methods employ a simple principle, i.e., the expansion rate of objects. Most expansion-based methods have been implemented for aerial robots (i.e., MAV and UAV). Such methods are fast, as they do not require extensive computations. However, they may fail when the surrounding objects become complex. Thus, in recent years, deep neural networks have been employed to meet the expectations of real-time applications. By default, expansion-based methods are fast. Thus, the detection accuracy in complex environments seems to be the primary concern of such techniques, which, by far, have not had much success.

4. Stereo-Based Obstacle Detection Techniques

Obstacle detection based on stereo uses two synchronised cameras fixed on the robot [36]. Figure 5 depicts a stereo camera pair mounted on a car to detect obstacles. These methods can be classified into two groups: IPM-based and disparity histogram-based [103].

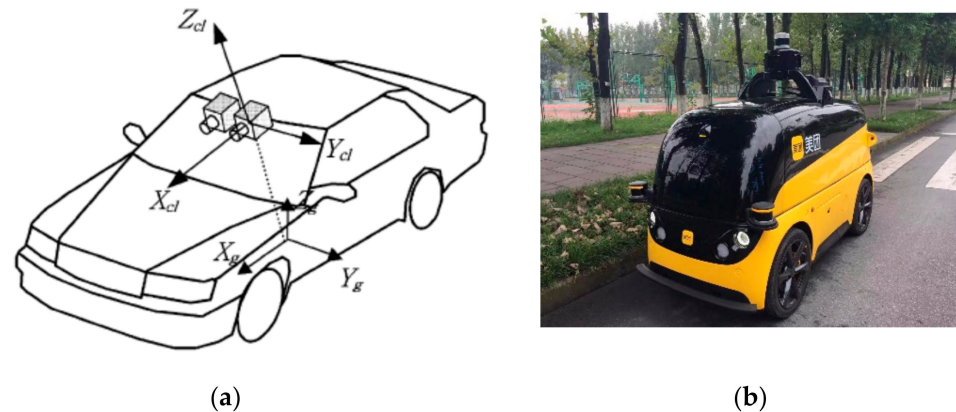


Figure 5. Stereo-based obstacle detection: (a) A pair of stereo cameras mounted on a car to detect an obstacle [36]; (b) Laser scanners and six cameras mounted on the autonomous vehicle at Southwest Jiaotong University take photos and point clouds in 360°.

4.1. IPM-Based Method

The IPM-based methods were primarily used to detect all types of road obstacles [104] and to eliminate the perspective effect of the original images in lane detection problems [105,106]. Currently, IPM images are mostly used in monocular methods [107–109].

Assuming the road has a flat surface, the IPM algorithm produces an image representing the road seen from the top, using internal and external parameters of the cameras. Then, the difference in the grey levels of pixels in the overlapping regions is computed, from which a polar histogram image is generated. If the image textures are uniform, this histogram contains two triangles/peaks: one for the lane and one for the potential obstacle. These peaks are then used to detect the obstacle, i.e., non-lane object. In effect, obstacle detection relies on identifying these two triangles based on their shapes and positions. In

practice, it becomes difficult to form such ideal triangles due to the diversity of textures in the images, objects of irregular shapes, and variations in the brightness of the pixels.

There is limited research on stereo IPM obstacle detection [110]. An example is that by Bertozzi et al. [110] for short-range obstacle detection. This method detects obstacles using the difference between the left and right IPM images. Although it may be accurate in some conditions, it has a limited range and cannot show the actual distance to obstacles. Kim et al. [111] used a stereo pair of cameras to create two IPM images for each camera. These images were then combined with another IPM image created using a pair of consecutive images taken with the camera having a smaller FOV to detect the obstacles.

Although IPM-based methods are very fast, they have two notable limitations. First, since they use object portions with uniform texture or colour for obstacle detection [112,113], they can only be used to detect objects like a car that has a uniform material [114]. Second, errors in the homography model, unknown camera motion, and light reflection from the floor can generate noise in the images [114]. Indeed, implementing the IPM transform requires a priori knowledge of the specific acquisition conditions (camera location, orientation, etc.) and some assumptions regarding the objects being imaged. Consequently, it can only be utilised in structured environments, where, for instance, the camera is fixed or when the system calibration and the surrounding environment can be monitored by another type of sensor [115]. Due to the limitations of this method, we recommend using it only for lane detection and obstacle detection in cars. This is because the necessary data and conditions for this method in unknown environments, particularly when using drones, are not necessarily available. Table 7 summarises the papers reviewed in this section.

Table 7. Recent papers on IPM-based obstacle detection methods.

Ref	Used for:	Principle	Strength(s)	Weakness(es)	NSOD	MOD	ODAD	FOD
[110]	Car	Use of the difference between the left and right IPM	Highly accurate	- limited range - Not calculating the precise distance	-	-	-	-
[111]	Mobile robot	Use of stereo cameras with two distinct FOVs	low-cost	Low-resolution images	-	-	-	-

4.2. Disparity Histogram-Based

Two cameras are installed at a fixed distance in front of the robot in these methods. The cameras have similar properties like focal length and FOV. They simultaneously capture two images of the surroundings. The acquired images are rectified. The distance between the matched pixels (disparity) is then calculated. This is repeated for all of the image pixels. The result is a disparity map, which is then used to compute the depth map of the surrounding objects [116]. Pixels having a depth smaller than a threshold are considered obstacle points. The majority of stereo-based obstacle detection techniques developed so far are disparity histogram-based which are reviewed in this section.

Disparity histogram-based methods can be discussed for robots on the ground and in the air. In the following, we will review both groups.

4.2.1. Disparity Histogram-Based Obstacle Detection for Terrestrial Robots

Disparity histogram-based obstacle detection techniques were initially developed for terrestrial robots. Kim et al. [117] proposed a Hierarchical Census Transform (HCT) matching method to develop car parking assistance using images taken by a pair of synchronised fisheye cameras. As the quality of points at the edges of a fisheye image is low, the detection

was only accurate enough in areas close to the image centre. Moreover, the algorithm's accuracy decreased when shadows and complicated or reflective backgrounds were present. Later, Ball et al. [118] introduced an obstacle detection algorithm that could continuously adapt to changes in the illumination and brightness in farm environments. They developed two distinct steps for obstacle detection. The first removes both the crop and the stubble. After that, stereo matching is performed only on the remaining small portions to increase the speed. The technique is unable to detect hidden obstacles. The second part bypasses this constraint by defining obstacles as unique observations in their appearance and structural cues [118]. Salhi and Amiri [119] proposed a faster algorithm implemented on Field Programmable Gate Arrays (FPGA) to simulate human visual systems.

Disparity histogram-based techniques rely on matching computationally intensive algorithms. A solution to speed up the computations is to reduce the matching search space. Jung et al. [103] and Huh et al. [36] removed the road pixels to reduce the search space and regarded the other pixels as obstacles for vehicles travelling along a road. To detect the road, they used the normal FOV cameras. As a result, only obstacles in front of the vehicle could be detected. In a similar approach, to guide visually impaired individuals, Huang et al. [120] used depth data obtained using a Kinect scanner to identify and remove the road points. Furthermore, Muhovič et al. [121] approximated the water surface by fitting a plane to the point cloud, and outlying points are processed further to identify potential obstacles. As a recent technique, Murmu and Nandi [122] presented a novel lane and obstacle detection algorithm that uses video frames captured by a low-cost stereo vision system. The suggested system generates a real-time disparity map from the sequential frames to identify lanes and other cars. Moreover, Sun et al. [123] used 3D point cloud candidates extracted by height analysis for obstacle detection instead of using all 3D point clouds.

With the development of neural networks, many researchers have recently turned their attention to deep learning methods [124–126]. Choe et al. [127] proposed a stereo object matching technique that uses 2D contextual information from images and 3D object-level information in the field of stereo matching. Luo et al. [128] also used CNNs that can produce extremely accurate results in less than one second. Moreover, in disparity histogram-based obstacle detection studies, Dairi et al. [124] developed a hybrid encoder that combines Deep Boltzmann Machines (DBM) and Auto-Encoders (AE). In addition, Song et al. [129] trained a convolutional neural network using manually labelled Region Of Interest (ROI) from the KITTI data set to classify the left/right side of the host lane. The 3-D data generated by stereo matching is used to generate an obstacle mask. Zhang et al. [125] introduced a method that uses stereo images and deep learning methods to avoid car accidents. The algorithm was developed for drivers reversing and with a limited view of the objects behind. This method detects and locates obstacles in the image using a faster R-CNN algorithm.

Haris and Hou [130] addressed how to improve the robustness of obstacle detection methods in a complex environment by integrating an MRF for obstacle detection, road segmentation, and the CNN model to navigate safely. Their research evaluated the detection of small obstacles left on the road [130]. Furthermore, Mukherjee et al. [131] provided a method for detecting and localising pedestrians using a ZED stereo camera. They used the Darknet YOLOv2 to locate and achieve more accurate and rapid obstacle detection results.

Compared with traditional methods, deep learning has the advantages of robustness, accuracy, and speed. In addition, it can achieve real-time, high-precision recognition and distance measurement through the combination of stereo vision techniques [125]. Table 8 provides a summary of the papers reviewed in this section. The majority of studies detected obstacles for cars. Additionally, the algorithms used in these studies used neural networks to reduce the matching search space and increase the accuracy of obstacle detection.

Table 8. Recent papers on disparity histogram-based obstacle detection techniques for terrestrial robots.

Ref	Used for:	Principle	Strength(s)	Weakness(es)	NSOD	MOD	ODAD	FOD
[120]	Blind people	Remove the surface and solve the problems of over-segmentation	Effective in a low light environment	Cannot be used outdoors	-	-	-	✓
[117]	Car	Presenting the HCT-based stereo matching method,	Real-time obstacle detection under a variety of lighting conditions	<ul style="list-style-type: none"> - The accuracy of obstacle detection, particularly for small obstacles, is limited by low-quality images of the fisheye - Difficult to use in situations with shadows, complicated backgrounds, and reflective surfaces 	-	✓	✓	-
[118]	Agricultural cars	Describes a vision-based obstacle detection system that continually adapts to environmental and illumination variations	Obstacles are detected in a variety of lighting conditions	Unable to detect obstacles that have been disguised in their appearance or structure	-	-	-	-
[119]	Car	Use of FPGA	Real-time	The communication between the modules is still to be optimised	-	-	-	✓
[121]	USV	Use of 3-D Point Cloud	Real-time	Problem of detecting large objects (e.g., coast and piers)	-	-	-	✓
[124]	Car	Use of unsupervised, deep-learning approaches	Real-time	Negative effect of high image noise on the method's performance	-	-	-	✓
[129]	Car	Use of convolutional neural network	lightweight stereo vision-based driving lane detection	Need appropriate training	-	-	-	✓
[125]	Car	Use of deep learning and binocular vision	Real-time and effective	Detecting pedestrians and ignoring other obstacles such as cars, trees, etc.	-	-	-	✓
[130]	Autonomous vehicle	Use of MRF	Small obstacle detection	Need to train the model on multiple training dataset	✓	-	-	✓
[122]	Self-driving car	Generating a disparity map	Real-time	low-resolution camera	-	-	-	✓
[131]	Car	Combining CNN and sensor fusion algorithms	Fast and credible results	Need appropriate training	-	-	-	✓
[123]	Car	Use of 3D point cloud candidates and a variant of RANSAC	Road obstacle detection such as potholes and cracks	Incomplete obstacle detection due to 3D point cloud candidates selection	-	-	-	✓

4.2.2. Disparity Histogram-Based Obstacle Detection for Aerial Robots

Despite terrestrial robots mainly being surrounded by known objects, aerial robots move in unknown environments. Processing disparity histogram-based methods can be too heavy for onboard MAV microprocessors. To simplify the search space and speed up the depth calculation, McGuire et al. [38] used vertical edges within the stereo images to

detect obstacles. Such an algorithm would not work in complicated environments where horizontal and diagonal edges are present.

Tijmons et al. [132] introduced the strategy of Droplet to identify and use only strong matched points and, thus, decrease the search space. In this study, the resolution of the images was reduced to increase the speed of disparity map generation. When the environment becomes complex, the processing speed of this technique decreases. Moreover, reducing the resolution may eliminate tiny obstacles such as tree branches, rope, and wire.

Barry et al. [4] concentrated only on fixed obstacles at a 5–10 m distance from a UAV to speed up the process. The algorithm was implemented on a light drone (less than 1 kg) and detected obstacles at 120 frames per second. The baseline of the cameras was only 14 inches. Thus, in addition to being limited to detecting fixed objects, a major challenge of this work would be its need for accurate calibration of the system to obtain reliable results. Lin et al. [133] considered dynamic environments and stereo cameras to detect moving obstacles using depth from stereo images. However, they only considered the obstacles' estimated position, size, and velocity. Therefore, some characteristics of objects such as direction, volume, shape, and influencing factors like environmental conditions were not considered. Hence, such algorithms may have difficulty detecting some of the moving obstacles.

One of the state-of-the-art methods is that by Grinberg and Ruf [134], which includes several components: image rectification, pixel matching, semi-global matching optimisation (SGM), compatibility check, and median filtering. This algorithm runs on an ARM processor of the Towards Ubiquitous Low-Power Image Processing Platforms (TULIPP). Therefore, image processing shows a performance suitable for real-time applications on a UAV [134].

The papers discussed in this section are summarised in Table 9. Overall, disparity histogram-based methods for ground and airborne mobile robots have common problems. One of them is computational complexity. To reduce computation time, the search space has to be reduced. In all the studies mentioned above, research has mainly focused on the speed issue by reducing the computing process or decreasing the resolution of the images. In this regard, removing the road segments has proven helpful for ground mobile robots and autonomous vehicles [22]. To reduce the number of computations, we propose combining disparity histogram-based and mono methods. This way, the mono method identifies the potential obstacles, while using the disparity histogram-based technique, the distance to those obstacles is estimated.

Table 9. Recent papers on disparity histogram-based obstacle detection techniques for aerial robots.

Ref	Used for:	Principle	Strength(s)	Weakness(es)	NSOD	MOD	ODAD	FOD
[38]	UAV	Computationally efficient optical flow and stereo algorithm	Reducing computational complexity to increase navigational efficiency	- Inefficiency in complex environments - Limited FOV	-	-	-	✓
[132]	MAV	Use of only strong matched points and reduction of images resolution	Fast	- Reducing image dimensions leads to the elimination of small obstacles - limited viewing angle	-	-	-	✓
[4]	UAV	Detecting only obstacles at a certain depth	Real-time	Instability due to the need for very exact camera calibration	-	-	-	✓
[133]	MAV	Depth estimation	Moving obstacle detection	The obstacle's velocity estimation may be very noisy	-	✓	-	-
[134]	UAV	Computation of the U-disparity and V-disparity maps	Real-time	FPGAs have limited memory capacity	-	-	-	✓

The second problem is sensitivity to illumination variations, which have been tackled using deep learning networks in recent years. The third problem is the accurate calibration of the stereo cameras. Suppose the stereo cameras are not calibrated correctly. The detection error increases very quickly over time. This is due to system instability which will affect the accuracy of computing the distance from the obstacle, especially when the baseline of the cameras is small, e.g., when they are mounted on small-sized UAVs.

4.3. Summary

In this section, stereo-based approaches were reviewed in two groups, i.e., IPM-based and Disparity histogram-based. They use a pair of synchronised images to detect obstacles. Despite their strengths, they suffer from limitations, making them less popular than monocular approaches. The IPM-based techniques work mainly in specific conditions and usually require prior knowledge to detect obstacles. Thus we only recommend using them for terrestrial applications like lane and obstacle detection to guide autonomous cars. They have a very difficult time when implemented on drones, as drones usually fly in unknown environments.

As for disparity histogram-based methods, a common problem is a computational complexity. To reduce computation time, the search space has to be reduced. This can be done by, for example, decreasing the resolution of the images or by removing the road segments (for ground mobile robots). To address this problem, we propose combining disparity histogram-based and monocular methods. This way, obstacles can be determined using monocular methods, while their depth can be obtained by disparity histogram-based techniques. The second problem is sensitivity to illumination variations, which deep learning networks have recently tackled. The third problem is the accurate calibration of the stereo cameras which should also be considered in developing a proper stereo-based obstacle detection algorithm.

5. Discussion and Prospects

In this paper, we discussed various obstacle detection methods. Nowadays, obstacle detection is a hot topic and a crucial part of most commercial unmanned vehicles. Almost every system under development includes an obstacle detection mechanism. Two types of techniques were reviewed: monocular and stereo. Monocular techniques, divided into appearance-based, motion-based, depth-based and expansion-based approaches, face different problems. Appearance-based methods are generally limited to environments where obstacles can easily be distinguished from the background. Motion-based obstacle detection relies mainly on the quality of the matching points. Indeed, detecting obstacles in front of the robot using optical flow is still an important challenge [75,76]. Moreover, most traditional depth-based methods require a complete map of the environment, which needs intensive computations. That is why deep learning-based depth estimation has widely been employed in recent years to improve speed. Compared to other monocular methods, expansion-based methods employ a simple principle, i.e., the expansion rate. Such methods are fast, as they do not require extensive computations. However, they may fail when the surrounding objects become complex. Thus, in recent years, deep neural networks have been employed to meet the expectations of real-time applications. By default, expansion-based methods are fast. Thus, the remaining problem is the accuracy of detecting obstacles in complex environments, which has been the primary concern, of course, with little success.

Stereo-based methods were also studied in two groups, i.e., IPM-based and Disparity histogram-based, the latter reviewed separately for ground and airborne mobile robots. Stereo techniques have almost common problems. One of them is computational complexity. In all the studies, research has mainly focused on the speed issue by reducing the computing process or decreasing the resolution of the images. In this regard, removing the road segments has proven helpful for ground mobile robots and autonomous vehicles. The second problem is sensitivity to illumination variations, which have been tackled using

deep learning networks in recent years. The third problem is the accurate calibration of the stereo cameras.

Based on this review, most algorithms' challenges are the ability to detect narrow, small, and moving obstacles in all directions and in real-time. Table 10 lists the papers we have found published since 2015 that address such abilities. As can be seen, the amount of research on detecting narrow and small obstacles is low, and, as such, detecting narrow and small obstacles is still a notable challenge. We believe that detecting these obstacles should be one of the areas to focus on in future research. Since the quality of detection increases with advancements in image resolution, we perceive that stronger algorithms will become available in the near future.

Table 10. The research overview of various methods of obstacle detection about the suggested criteria.

Criteria		NSOD	MOD	ODAD	FOD
Methods					
Monocular	Appearance-based	[32]	[30,52,59]	[58]	[52,57,59,60,63,65–67,135,136]
	Motion-based	-	-	-	[70]
	Depth-based	-	-	[83,84,86]	[76,87]
	Expansion-based	-	[69]	-	[5]
	Stereo	[4,130]	[117,133]	[118]	[4,38,119–125,129,131,132,134]

Additionally, the ability for real-time or high-speed obstacle detection should be one of the key areas of future research. Indeed the development of fast obstacle detection systems for unmanned vehicles is now more needed than ever. For instance, self-driving automobiles need to constantly detect obstacles such as pedestrians, other vehicles, bicycles, and animals at an incredibly fast rate to ensure driving safety. The obstacle detection speed is also crucial for the fast-moving UAVs within cluttered areas like forest canopies. An option for achieving this is to use deep learning. Another option is to develop parallel processing hardware and software, which, in recent years, have shown could substantially enhance the processing speed. Through the review, we realised that single-image methods are the fastest methods due to their low computational complexity. It appears that their performance will even increase in the future as extremely fast deep learning algorithms continue to evolve.

One notable finding was that most techniques employ cameras with FOVs ranging from 57° to 92°. This angle of view covers only a portion of the area in front of a robot. As a result, obstacles beyond the camera's FOV cannot be identified. To deal with this problem, as previously stated, there have been systems that use fisheye cameras to detect obstacles approaching the robot from any direction. A solution is to employ a monocular or stereo-based obstacle detection algorithm that uses a single/pair of synchronised fisheye cameras. However, as with MAVs or UAVs, if the stereo baseline is small, very precise calibration of the system is required, or the system becomes unstable over time. This means that implementing a stereo fisheye obstacle detection algorithm on flying robots is a challenging task that should be closely examined and/or explored in future studies. Moreover, robots utilising fisheye cameras have difficulty dealing with objects projected close to the sides of the fisheye image.

Another issue that should be considered in future research is detecting fast-moving objects, especially in crowded areas. An application of this is the development of an algorithm to detect moving cars, humans, and bicycles in urban areas for a self-driving car.

In Table 11, image-based methods are compared based on accuracy, speed and cost. In terms of accuracy, expansion-based approaches can be very accurate, as recent research on this method has focused on fully obstacle detection. In addition, if the system calibration is accurate, stereo methods will be accurate. Appearance-based and expansion-based approaches are faster since their calculations are simple and less expensive. In addition, all monocular approaches are cost-effective due to the use of one camera.

Table 11. Comparison of image-based methods based on accuracy, speed and cost. Asterisk symbols indicate the effectiveness of the method.

Criteria		Accuracy	Speed	Cost
Methods				
Monocular	Appearance-based	*	***	**
	Motion-based	*	*	**
	Depth-based	**	*	**
	Expansion-based	***	***	**
	Stereo	***	*	*

6. Conclusions and Suggestions for Future Works

This paper explored and reviewed image-based obstacle detection techniques implemented mainly on UAVs and autonomous vehicles. Two groups of algorithms/systems were reviewed: monocular and stereo. Monocular algorithms were discussed in appearance-based, motion-based, depth-based, and expansion-based sub-groups, while stereo techniques were reviewed separately for IPM-based and disparity histogram-based methods.

Monocular-based obstacle detection approaches use only one camera. They have simple computations and are fast. As a result, researchers have implemented/investigated these methods extensively for aerial and terrestrial robot navigation. Due to the small size, low processing power, and weight limitations of MAVs or small UAVs, most studies have used monocular methods. On the other hand, stereo-based methods use a pair(s) of synchronised cameras to capture images for generating a 3D map from the surrounding objects in real-time. The map is then used to locate the obstacle. Unfortunately, such methods are not computationally cost-effective [4,38]. To resolve this issue, researchers usually require a powerful Graphics Processing Unit (GPU) [99].

Moreover, they should rely on accurate system calibration. This is because any error can affect the system's stability over time [4]. Nevertheless, one of the primary benefits of these methods is calculating the distance between the robot and the obstacle.

In recent years, the research has focused mainly on using deep learning networks for fast and accurate obstacle detection. We suggest obtaining the training data from publicly available datasets to generate dense probabilistic semantic maps in urban environments automatically. This can provide robust labels for environmental objects such as roads, trees, buildings, etc.

There has been significant progress in increasing the speed of computations, though, in terms of accuracy; there is still a long way to go, especially for complex environments where objects of varying types and complexities are present. Moreover, the review suggests that detecting narrow and small and moving obstacles are the most challenging problems to focus on in future studies. Detecting such obstacles is particularly important for aerial robots.

Author Contributions: Conceptualization, S.B., M.V., and S.P.; methodology, S.B., M.V., and S.P.; investigation, S.B., M.V., S.P. and J.L.; resources S.B., M.V., S.P. and J.L.; software, S.B., M.V., S.P. and J.L.; writing—original draft preparation, S.B., M.V., and S.P.; writing—review and editing, S.B., M.V., S.P. and J.L.; funding acquisition, S.B., M.V., S.P.;. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: No institutional review applied.

Informed Consent Statement: No applicable.

Data Availability Statement: No data and code available.

Acknowledgments: The authors appreciate the technical research support of the GeoAI Smarter Map and LiDAR Lab of the Faculty of Geosciences and Environmental Engineering, Southwest Jiaotong University (SWJTU). This work results from a joint research study between K.N. Toosi University of Technology (KNT), Iran, and the University of Waterloo (UW), Canada. We also thank Saeid Homayouni from Centre Eau Terre Environnement, Institut National de la Recherche Scientifique, Québec, Canada, for his kind support and contribution to the improvement of this paper.

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

References

- Colomina, I.; Molina, P. Unmanned aerial systems for photogrammetry and remote sensing: A review. *ISPRS J. Photogramm. Remote Sens.* **2014**, *92*, 79–97. [[CrossRef](#)]
- Reinoso, J.; Gonçalves, J.; Pereira, C.; Bleninger, T. Cartography for Civil Engineering Projects: Photogrammetry Supported by Unmanned Aerial Vehicles. *Iran. J. Sci. Technol. Trans. Civ. Eng.* **2018**, *42*, 91–96. [[CrossRef](#)]
- Janoušek, J.; Jambor, V.; Marcoň, P.; Dohnal, P.; Synková, H.; Fiala, P. Using UAV-Based Photogrammetry to Obtain Correlation between the Vegetation Indices and Chemical Analysis of Agricultural Crops. *Remote Sens.* **2021**, *13*, 1878. [[CrossRef](#)]
- Barry, A.J.; Florence, P.R.; Tedrake, R. High-speed autonomous obstacle avoidance with pushbroom stereo. *J. Field Robot.* **2018**, *35*, 52–68. [[CrossRef](#)]
- Lee, H.; Ho, H.; Zhou, Y. Deep Learning-based Monocular Obstacle Avoidance for Unmanned Aerial Vehicle Navigation in Tree Plantations. *J. Intell. Robot. Syst.* **2021**, *101*, 5. [[CrossRef](#)]
- Toth, C.; Józków, G. Remote sensing platforms and sensors: A survey. *ISPRS J. Photogramm. Remote Sens.* **2016**, *115*, 22–36. [[CrossRef](#)]
- Goodin, C.; Carrillo, J.; Monroe, J.G.; Carruth, D.W.; Hudson, C.R. An Analytic Model for Negative Obstacle Detection with Lidar and Numerical Validation Using Physics-Based Simulation. *Sensors* **2021**, *21*, 3211. [[CrossRef](#)] [[PubMed](#)]
- Hu, J.-w.; Zheng, B.-y.; Wang, C.; Zhao, C.-h.; Hou, X.-l.; Pan, Q.; Xu, Z. A survey on multi-sensor fusion based obstacle detection for intelligent ground vehicles in off-road environments. *Front. Inf. Technol. Electron. Eng.* **2020**, *21*, 675–692. [[CrossRef](#)]
- John, V.; Mita, S. Deep Feature-Level Sensor Fusion Using Skip Connections for Real-Time Object Detection in Autonomous Driving. *Electronics* **2021**, *10*, 424. [[CrossRef](#)]
- Serna, A.; Marcotegui, B. Urban accessibility diagnosis from mobile laser scanning data. *ISPRS J. Photogramm. Remote Sens.* **2013**, *84*, 23–32. [[CrossRef](#)]
- Díaz-Vilariño, L.; Boguslawski, P.; Khoshelham, K.; Lorenzo, H.; Mahdjoubi, L. INDOOR NAVIGATION FROM POINT CLOUDS: 3D MODELLING AND OBSTACLE DETECTION. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *41*, 275–281. [[CrossRef](#)]
- Li, F.; Wang, H.; Akwensi, P.H.; Kang, Z. Construction of Obstacle Element Map Based on Indoor Scene Recognition. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2019**, *XLII-2/W13*, 819–825. [[CrossRef](#)]
- Keramatian, A.; Gulisano, V.; Papatriantafidou, M.; Tsigas, P. Mad-c: Multi-stage approximate distributed cluster-combining for obstacle detection and localization. *J. Parallel Distrib. Comput.* **2021**, *147*, 248–267. [[CrossRef](#)]
- Gianni, C.; Balsi, M.; Esposito, S.; Fallavollita, P. Obstacle Detection System Involving Fusion of Multiple Sensor Technologies. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *42*, 127–134. [[CrossRef](#)]
- Xie, G.; Zhang, J.; Tang, J.; Zhao, H.; Sun, N.; Hu, M. Obstacle detection based on depth fusion of lidar and radar in challenging conditions. In *Industrial Robot: The International Journal of Robotics Research and Application*; Emerald Group Publishing Limited: Bradford, UK, 2021. [[CrossRef](#)]
- Qin, R.; Zhao, X.; Zhu, W.; Yang, Q.; He, B.; Li, G.; Yan, T. Multiple Receptive Field Network (MRF-Net) for Autonomous Underwater Vehicle Fishing Net Detection Using Forward-Looking Sonar Images. *Sensors* **2021**, *21*, 1933. [[CrossRef](#)]
- Yılmaz, E.; Özyer, S.T. Remote and Autonomous Controlled Robotic Car based on Arduino with Real Time Obstacle Detection and Avoidance. *Univers. J. Eng. Sci.* **2019**, *7*, 1–7. [[CrossRef](#)]
- Singh, B.; Kapoor, M. A Framework for the Generation of Obstacle Data for the Study of Obstacle Detection by Ultrasonic Sensors. *IEEE Sens. J.* **2021**, *21*, 9475–9483. [[CrossRef](#)]
- Kucukyildiz, G.; Ocak, H.; Karakaya, S.; Sayli, O. Design and implementation of a multi sensor based brain computer interface for a robotic wheelchair. *J. Intell. Robot. Syst.* **2017**, *87*, 247–263. [[CrossRef](#)]
- Quaglia, G.; Visconte, C.; Scimmi, L.S.; Melchiorre, M.; Cavallone, P.; Pastorelli, S. Design of a UGV powered by solar energy for precision agriculture. *Robotics* **2020**, *9*, 13. [[CrossRef](#)]
- Pirasteh, S.; Shamsipour, G.; Liu, G.; Zhu, Q.; Chengming, Y. A new algorithm for landslide geometric and deformation analysis supported by digital elevation models. *Earth Sci. Inform.* **2020**, *13*, 361–375. [[CrossRef](#)]
- Ye, C.; Li, H.; Wei, R.; Wang, L.; Sui, T.; Bai, W.; Saied, P. Double Adaptive Intensity-Threshold Method for Uneven Lidar Data to Extract Road Markings. *Photogramm. Eng. Remote Sens.* **2021**, *87*, 639–648. [[CrossRef](#)]
- Li, H.; Ye, W.; Liu, J.; Tan, W.; Pirasteh, S.; Fathollahi, S.N.; Li, J. High-Resolution Terrain Modeling Using Airborne LiDAR Data with Transfer Learning. *Remote Sens.* **2021**, *13*, 3448. [[CrossRef](#)]
- Ghasemi, M.; Varshosaz, M.; Pirasteh, S.; Shamsipour, G. Optimizing Sector Ring Histogram of Oriented Gradients for human injured detection from drone images. *Geomat. Nat. Hazards Risk* **2021**, *12*, 581–604. [[CrossRef](#)]

25. Yazdan, R.; Varshosaz, M.; Pirasteh, S.; Remondino, F. Using geometric constraints to improve performance of image classifiers for automatic segmentation of traffic signs. *Geomatica* **2021**, *75*, 28–50. [[CrossRef](#)]
26. Al-Obaidi, A.S.M.; Al-Qassar, A.; Nasser, A.R.; Alkhayyat, A.; Humaidi, A.J.; Ibraheem, I.K. Embedded design and implementation of mobile robot for surveillance applications. *Indones. J. Sci. Technol.* **2021**, *6*, 427–440. [[CrossRef](#)]
27. Foroutan, M.; Tian, W.; Goodin, C.T. Assessing impact of understory vegetation density on solid obstacle detection for off-road autonomous ground vehicles. *ASME Lett. Dyn. Syst. Control.* **2021**, *1*, 021008. [[CrossRef](#)]
28. Han, S.; Jiang, Y.; Bai, Y. Fast-PGMED: Fast and Dense Elevation Determination for Earthwork Using Drone and Deep Learning. *J. Constr. Eng. Manag.* **2022**, *148*, 04022008. [[CrossRef](#)]
29. Tondin Ferreira Dias, E.; Vieira Neto, H.; Schneider, F.K. A Compressed Sensing Approach for Multiple Obstacle Localisation Using Sonar Sensors in Air. *Sensors* **2020**, *20*, 5511. [[CrossRef](#)]
30. Huh, S.; Cho, S.; Jung, Y.; Shim, D.H. Vision-based sense-and-avoid framework for unmanned aerial vehicles. *IEEE Trans. Aerosp. Electron. Syst.* **2015**, *51*, 3427–3439. [[CrossRef](#)]
31. Aswini, N.; Krishna Kumar, E.; Uma, S. UAV and obstacle sensing techniques—a perspective. *Int. J. Intell. Unmanned Syst.* **2018**, *6*, 32–46. [[CrossRef](#)]
32. Lee, T.-J.; Yi, D.-H.; Cho, D.-I. A monocular vision sensor-based obstacle detection algorithm for autonomous robots. *Sensors* **2016**, *16*, 311. [[CrossRef](#)] [[PubMed](#)]
33. Zahran, S.; Moussa, A.M.; Sesay, A.B.; El-Sheimy, N. A new velocity meter based on Hall effect sensors for UAV indoor navigation. *IEEE Sens. J.* **2018**, *19*, 3067–3076. [[CrossRef](#)]
34. Mashaly, A.S.; Wang, Y.; Liu, Q. Efficient sky segmentation approach for small UAV autonomous obstacles avoidance in cluttered environment. In Proceedings of the Geoscience and Remote Sensing Symposium (IGARSS), 2016 IEEE International, Beijing, China, 10–15 July 2016; pp. 6710–6713.
35. Al-Kaff, A.; García, F.; Martín, D.; De La Escalera, A.; Armingol, J.M. Obstacle detection and avoidance system based on monocular camera and size expansion algorithm for UAVs. *Sensors* **2017**, *17*, 1061. [[CrossRef](#)]
36. Huh, K.; Park, J.; Hwang, J.; Hong, D. A stereo vision-based obstacle detection system in vehicles. *Opt. Lasers Eng.* **2008**, *46*, 168–178. [[CrossRef](#)]
37. Padhy, R.P.; Choudhury, S.K.; Sa, P.K.; Bakshi, S. Obstacle Avoidance for Unmanned Aerial Vehicles: Using Visual Features in Unknown Environments. *IEEE Consum. Electron. Mag.* **2019**, *8*, 74–80. [[CrossRef](#)]
38. McGuire, K.; de Croon, G.; De Wagter, C.; Tuyls, K.; Kappen, H.J. Efficient Optical Flow and Stereo Vision for Velocity Estimation and Obstacle Avoidance on an Autonomous Pocket Drone. *IEEE Robot. Autom. Lett.* **2017**, *2*, 1070–1076. [[CrossRef](#)]
39. Sun, B.; Li, W.; Liu, H.; Yan, J.; Gao, S.; Feng, P. Obstacle Detection of Intelligent Vehicle Based on Fusion of Lidar and Machine Vision. *Eng. Lett.* **2021**, *29*, EL_29_2_41.
40. Ristić-Durrant, D.; Franke, M.; Michels, K. A Review of Vision-Based On-Board Obstacle Detection and Distance Estimation in Railways. *Sensors* **2021**, *21*, 3452. [[CrossRef](#)]
41. Nobile, L.; Randazzo, M.; Colledanchise, M.; Monorchio, L.; Villa, W.; Puja, F.; Natale, L. Active Exploration for Obstacle Detection on a Mobile Humanoid Robot. *Actuators* **2021**, *10*, 205. [[CrossRef](#)]
42. Yu, X.; Marinov, M. A study on recent developments and issues with obstacle detection systems for automated vehicles. *Sustainability* **2020**, *12*, 3281. [[CrossRef](#)]
43. Yeong, D.J.; Velasco-Hernandez, G.; Barry, J.; Walsh, J. Sensor and sensor fusion technology in autonomous vehicles: A review. *Sensors* **2021**, *21*, 2140. [[CrossRef](#)] [[PubMed](#)]
44. Nieuwenhuisen, M.; Droeschel, D.; Schneider, J.; Holz, D.; Labe, T.; Behnke, S. Multimodal obstacle detection and collision avoidance for micro aerial vehicles. In Proceedings of the Mobile Robots (ECMR), 2013 European Conference on Mobile Robots, Residència d’Investigadors, Barcelona, Spain, 25 September 2013; pp. 7–12.
45. Droeschel, D.; Nieuwenhuisen, M.; Beul, M.; Holz, D.; Stückler, J.; Behnke, S. Multilayered mapping and navigation for autonomous micro aerial vehicles. *J. Field Robot.* **2016**, *33*, 451–475. [[CrossRef](#)]
46. D’souza, M.M.; Agrawal, A.; Tina, V.; HR, V.; Navya, T. Autonomous Walking with Guiding Stick for the Blind Using Echolocation and Image Processing. *Methodology* **2019**, *7*, 66–71. [[CrossRef](#)]
47. Carrio, A.; Lin, Y.; Saripalli, S.; Campoy, P. Obstacle detection system for small UAVs using ADS-B and thermal imaging. *J. Intell. Robot. Syst.* **2017**, *88*, 583–595. [[CrossRef](#)]
48. Beul, M.; Krombach, N.; Nieuwenhuisen, M.; Droeschel, D.; Behnke, S. Autonomous navigation in a warehouse with a cognitive micro aerial vehicle. In *Robot Operating System (ROS)*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 487–524.
49. John, V.; Nithilan, M.; Mita, S.; Tehrani, H.; Sudheesh, R.; Lalu, P. So-net: Joint semantic segmentation and obstacle detection using deep fusion of monocular camera and radar. In Proceedings of the Pacific-Rim Symposium on Image and Video Technology, Sydney, NSW, Australia, 18–22 November 2019; pp. 138–148.
50. Kragh, M.; Underwood, J. Multi-Modal Obstacle Detection in Unstructured Environments with Conditional Random Fields. *J. Field Robot.* **2017**, *37*, 53–72. [[CrossRef](#)]
51. Singh, Y.; Kaur, L. Obstacle Detection Techniques in Outdoor Environment: Process, Study and Analysis. *Int. J. Image Graph. Signal Processing* **2017**, *9*, 35–53. [[CrossRef](#)]
52. Qiu, Z.; Zhao, N.; Zhou, L.; Wang, M.; Yang, L.; Fang, H.; He, Y.; Liu, Y. Vision-based moving obstacle detection and tracking in paddy field using improved yolov3 and deep SORT. *Sensors* **2020**, *20*, 4082. [[CrossRef](#)]

53. Gharani, P.; Karimi, H.A. Context-aware obstacle detection for navigation by visually impaired. *Image Vis. Comput.* **2017**, *64*, 103–115. [[CrossRef](#)]
54. Hatch, K.; Mern, J.M.; Kochenderfer, M.J. Obstacle Avoidance Using a Monocular Camera. In Proceedings of the AIAA Scitech 2021 Forum, Virtual Event, 11–22 January 2021; p. 0269.
55. Badrloo, S.; Varshosaz, M. Monocular vision based obstacle detection. *Earth Obs. Geomat. Eng.* **2017**, *1*, 122–130. [[CrossRef](#)]
56. Ulrich, I.; Nourbakhsh, I. Appearance-based obstacle detection with monocular color vision. In Proceedings of the AAAI/IAAI, Austin, TX, USA, 30 July 2000; pp. 866–871.
57. Liu, J.; Li, H.; Liu, J.; Xie, S.; Luo, J. Real-Time Monocular Obstacle Detection Based on Horizon Line and Saliency Estimation for Unmanned Surface Vehicles. *Mob. Netw. Appl.* **2021**, *26*, 1372–1385. [[CrossRef](#)]
58. Shih An, L.; Chou, L.-H.; Chang, T.-H.; Yang, C.-H.; Chang, Y.-C. Obstacle Avoidance of Mobile Robot Based on HyperOmni Vision. *Sens. Mater.* **2019**, *31*, 1021. [[CrossRef](#)]
59. Wang, S.-H.; Li, X.-X. A Real-Time Monocular Vision-Based Obstacle Detection. In Proceedings of the 2020 6th International Conference on Control, Automation and Robotics (ICCAR), Singapore, 20–23 April 2020; pp. 695–699.
60. Talele, A.; Patil, A.; Barse, B. Detection of real time objects using TensorFlow and OpenCV. *Asian J. Converg. Technol. (AJCT)* **2019**, *5*.
61. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M. {TensorFlow}: A System for {Large-Scale} Machine Learning. In Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), Savannah, GA, USA, 2–4 November 2016; pp. 265–283.
62. Bradski, G. The openCV library. In *Dr. Dobb's Journal: Software Tools for the Professional Programmer*; M & T Pub., the University of Michigan: Ann Arbor, MI, USA, 2000; Volume 25, pp. 120–123.
63. Rane, M.; Patil, A.; Barse, B. Real object detection using TensorFlow. In *ICCCE 2019*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 39–45.
64. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
65. He, D.; Zou, Z.; Chen, Y.; Liu, B.; Yao, X.; Shan, S. Obstacle detection of rail transit based on deep learning. *Measurement* **2021**, *176*, 109241. [[CrossRef](#)]
66. He, Y.; Liu, Z. A Feature Fusion Method to Improve the Driving Obstacle Detection under Foggy Weather. *IEEE Trans. Transp. Electrif.* **2021**, *7*, 2505–2515. [[CrossRef](#)]
67. Liu, J.; Li, H.; Luo, J.; Xie, S.; Sun, Y. Efficient obstacle detection based on prior estimation network and spatially constrained mixture model for unmanned surface vehicles. *J. Field Robot.* **2021**, *38*, 212–228. [[CrossRef](#)]
68. de Croon, G.; De Wagter, C. Learning what is above and what is below: Horizon approach to monocular obstacle detection. *arXiv* **2018**, arXiv:1806.08007.
69. Zeng, Y.; Zhao, F.; Wang, G.; Zhang, L.; Xu, B. Brain-Inspired Obstacle Detection Based on the Biological Visual Pathway. In Proceedings of the International Conference on Brain and Health Informatics, Omaha, NE, USA, 13–16 October 2016; pp. 355–364.
70. Jia, B.; Liu, R.; Zhu, M. Real-time obstacle detection with motion features using monocular vision. *Vis. Comput.* **2015**, *31*, 281–293. [[CrossRef](#)]
71. Ohnishi, N.; Imiya, A. Appearance-based navigation and homing for autonomous mobile robot. *Image Vis. Comput.* **2013**, *31*, 511–532. [[CrossRef](#)]
72. Tsai, C.-C.; Chang, C.-W.; Tao, C.-W. Vision-Based Obstacle Detection for Mobile Robot in Outdoor Environment. *J. Inf. Sci. Eng.* **2018**, *34*, 21–34.
73. Gunn, S.R. Support vector machines for classification and regression. *ISIS Tech. Rep.* **1998**, *14*, 5–16.
74. Bay, H.; Ess, A.; Tuytelaars, T.; Van Gool, L. Speeded-up robust features (SURF). *Comput. Vis. Image Underst.* **2008**, *110*, 346–359. [[CrossRef](#)]
75. de Croon, G.; De Wagter, C.; Seidl, T. Enhancing optical-flow-based control by learning visual appearance cues for flying robots. *Nat. Mach. Intell.* **2021**, *3*, 33–41. [[CrossRef](#)]
76. Urban, D.; Caplier, A. Time- and Resource-Efficient Time-to-Collision Forecasting for Indoor Pedestrian Obstacles Avoidance. *J. Imaging* **2021**, *7*, 61. [[CrossRef](#)] [[PubMed](#)]
77. Nalpantidis, L.; Gasteratos, A. Stereo vision depth estimation methods for robotic applications. In *Depth Map and 3D Imaging Applications: Algorithms and Technologies*; IGI Global: Hershey/Derry/Dauphin, PA, USA, 2012; pp. 397–417.
78. Lee, J.; Jeong, J.; Cho, J.; Yoo, D.; Lee, B.; Lee, B. Deep neural network for multi-depth hologram generation and its training strategy. *Opt. Express* **2020**, *28*, 27137–27154. [[CrossRef](#)] [[PubMed](#)]
79. Almalioglu, Y.; Saputra, M.R.U.; De Gusmao, P.P.; Markham, A.; Trigoni, N. GANVO: Unsupervised deep monocular visual odometry and depth estimation with generative adversarial networks. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Colombo, Sri Lanka, 9–10 May 2019; pp. 5474–5480.
80. Kim, D.; Ga, W.; Ahn, P.; Joo, D.; Chun, S.; Kim, J. Global-Local Path Networks for Monocular Depth Estimation with Vertical CutDepth. *arXiv* **2022**, arXiv:2201.07436.
81. Gao, W.; Wang, K.; Ding, W.; Gao, F.; Qin, T.; Shen, S. Autonomous aerial robot using dual-fisheye cameras. *J. Field Robot.* **2020**, *37*, 497–514. [[CrossRef](#)]
82. Silva, A.; Mendonça, R.; Santana, P. Monocular Trail Detection and Tracking Aided by Visual SLAM for Small Unmanned Aerial Vehicles. *J. Intell. Robot. Syst.* **2020**, *97*, 531–551. [[CrossRef](#)]

83. Häne, C.; Heng, L.; Lee, G.H.; Fraundorfer, F.; Furgale, P.; Sattler, T.; Pollefeys, M. 3D visual perception for self-driving cars using a multi-camera system: Calibration, mapping, localization, and obstacle detection. *Image Vis. Comput.* **2017**, *68*, 14–27. [[CrossRef](#)]
84. Lin, Y.; Gao, F.; Qin, T.; Gao, W.; Liu, T.; Wu, W.; Yang, Z.; Shen, S. Autonomous aerial navigation using monocular visual-inertial fusion. *J. Field Robot.* **2018**, *35*, 23–51. [[CrossRef](#)]
85. Zhao, C.; Sun, Q.; Zhang, C.; Tang, Y.; Qian, F. Monocular depth estimation based on deep learning: An overview. *Sci. China Technol. Sci.* **2020**, *63*, 1612–1627. [[CrossRef](#)]
86. Kumar, V.R.; Milz, S.; Simon, M.; Witt, C.; Amende, K.; Petzold, J.; Yogamani, S. Monocular Fisheye Camera Depth Estimation Using Semi-supervised Sparse Velodyne Data. *arXiv* **2018**, arXiv:1803.06192. [[CrossRef](#)]
87. Mancini, M.; Costante, G.; Valigi, P.; Ciarruglia, T.A. J-MOD 2: Joint Monocular Obstacle Detection and Depth Estimation. *IEEE Robot. Autom. Lett.* **2018**, *3*, 1490–1497. [[CrossRef](#)]
88. Haseeb, M.A.; Guan, J.; Ristic-Durrant, D.; Gräser, A. DisNet: A novel method for distance estimation from monocular camera. In Proceedings of the 10th Planning, Perception and Navigation for Intelligent Vehicles (PPNIV18), IROS, Madrid, Spain, 1 October 2018.
89. Mátyus, G.; Luo, W.; Urtasun, R. Deeproadmapper: Extracting road topology from aerial images. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 3438–3446.
90. Homayounfar, N.; Ma, W.-C.; Liang, J.; Wu, X.; Fan, J.; Urtasun, R. Dagmapper: Learning to map by discovering lane topology. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Long Beach, CA, USA, 16–17 June 2019; pp. 2911–2920.
91. Maturana, D.; Chou, P.-W.; Uenoyama, M.; Scherer, S. Real-time semantic mapping for autonomous off-road navigation. In Proceedings of the Field and Service Robotics, Toronto, ON, Canada, 25 April 2018; pp. 335–350.
92. Sengupta, S.; Sturgess, P.; Ladický, L.u.; Torr, P.H. Automatic dense visual semantic mapping from street-level imagery. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Algarve, Portugal, 7–12 October 2012; pp. 857–862.
93. Gerke, M.; Xiao, J. Fusion of airborne laserscanning point clouds and images for supervised and unsupervised scene classification. *ISPRS J. Photogramm. Remote Sens.* **2014**, *87*, 78–92. [[CrossRef](#)]
94. Seif, H.G.; Hu, X. Autonomous driving in the iCity—HD maps as a key challenge of the automotive industry. *Engineering* **2016**, *2*, 159–162. [[CrossRef](#)]
95. Jiao, J. Machine learning assisted high-definition map creation. In Proceedings of the 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC), Tokyo, Japan, 23–27 July 2018; pp. 367–373.
96. Ye, C.; Zhao, H.; Ma, L.; Jiang, H.; Li, H.; Wang, R.; Chapman, M.A.; Junior, J.M.; Li, J. Robust lane extraction from MLS point clouds towards HD maps especially in curve road. *IEEE Trans. Intell. Transp. Syst.* **2020**, *23*, 1505–1518. [[CrossRef](#)]
97. Ming, Y.; Meng, X.; Fan, C.; Yu, H. Deep learning for monocular depth estimation: A review. *Neurocomputing* **2021**, *438*, 14–33. [[CrossRef](#)]
98. Mori, T.; Scherer, S. First results in detecting and avoiding frontal obstacles from a monocular camera for micro unmanned aerial vehicles. In Proceedings of the Robotics and Automation (Icra), 2013 IEEE International Conference on, Karlsruhe, Germany, 6–10 May 2013; pp. 1750–1757.
99. Aguilar, W.G.; Casaliglla, V.P.; Pólit, J.L. Obstacle avoidance based-visual navigation for micro aerial vehicles. *Electronics* **2017**, *6*, 10. [[CrossRef](#)]
100. Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [[CrossRef](#)]
101. Escobar-Alvarez, H.D.; Johnson, N.; Hebble, T.; Klingebiel, K.; Quintero, S.A.; Regenstein, J.; Browning, N.A. R-ADVANCE: Rapid Adaptive Prediction for Vision-based Autonomous Navigation, Control, and Evasion. *J. Field Robot.* **2018**, *35*, 91–100. [[CrossRef](#)]
102. Badrloo, S.; Varshosaz, M.; Pirasteh, S.; Li, J. A novel region-based expansion rate obstacle detection method for MAVs using a fisheye camera. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *108*, 102739. [[CrossRef](#)]
103. Jung, H.; Lee, Y.; Kim, B.; Yoon, P.; Kim, J. Stereo vision-based forward obstacle detection. *Int. J. Automot. Technol.* **2007**, *8*, 493–504.
104. Parmar, M.M.; Rawlo, R.R.; Shirke, J.L.; Sangam, S. Self-Driving Car. *Int. J. Res. Appl. Sci. Eng. Technol. (IJRASET)*. **2022**, *10*, 2305–2309. [[CrossRef](#)]
105. Wang, C.; Shi, Z.-k. A novel traffic stream detection method based on inverse perspective mapping. *Procedia Eng.* **2012**, *29*, 1938–1943. [[CrossRef](#)]
106. Muad, A.M.; Hussain, A.; Samad, S.A.; Mustafa, M.M.; Majlis, B.Y. Implementation of inverse perspective mapping algorithm for the development of an automatic lane tracking system. In Proceedings of the 2004 IEEE Region 10 Conference TENCON, Chiang Mai, Thailand, 21–24 November 2004; pp. 207–210.
107. Kuo, L.-C.; Tai, C.-C. Robust Image-Based Water-Level Estimation Using Single-Camera Monitoring. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 1–11. [[CrossRef](#)]
108. Hu, Z.; Xiao, H.; Zhou, Z.; Li, N. Detection of parking slots occupation by temporal difference of inverse perspective mapping from vehicle-borne monocular camera. *Proc. Inst. Mech. Eng. Part D J. Automob. Eng.* **2021**, *235*, 3119–3126. [[CrossRef](#)]
109. Lin, C.-T.; Shen, T.-K.; Shou, Y.-W. Construction of fisheye lens inverse perspective mapping model and its applications of obstacle detection. *EURASIP J. Adv. Signal Processing* **2010**, *2010*, 296598. [[CrossRef](#)]
110. Bertozzi, M.; Broggi, A. GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection. *IEEE Trans. Image Processing* **1998**, *7*, 62–81. [[CrossRef](#)] [[PubMed](#)]

111. Kim, C.-h.; Lee, T.-j. An application of stereo camera with two different FoVs for SLAM and obstacle detection. *IFAC-PapersOnLine* **2018**, *51*, 148–153. [[CrossRef](#)]
112. Wang, H.; Yuan, K.; Zou, W.; Peng, Y. Real-time region-based obstacle detection with monocular vision. In Proceedings of the 2005 IEEE International Conference on Robotics and Biomimetics-ROBIO, Hong Kong, China, 5–9 July 2005; pp. 615–619.
113. Fazl-Ersi, E.; Tsotsos, J.K. Region classification for robust floor detection in indoor environments. In Proceedings of the International Conference Image Analysis and Recognition, Halifax, NS, Canada, 6–8 July 2009; pp. 717–726.
114. Cucchiara, R.; Perini, E.; Pistoni, G. Efficient Stereo Vision for Obstacle Detection and AGV Navigation. In Proceedings of the ICIAP, Modena, Italy, 10–14 September 2007; pp. 291–296.
115. Tanveer, M.H.; Sgorbissa, A. An Inverse Perspective Mapping Approach using Monocular Camera of Pepper Humanoid Robot to Determine the Position of Other Moving Robot in Plane. In Proceedings of the ICINCO (2), Porto, Portugal, 29–31 July 2018; pp. 229–235.
116. Song, W.; Xiong, G.; Cao, L.; Jiang, Y. Depth calculation and object detection using stereo vision with subpixel disparity and hog feature. In *Advances in Information Technology and Education*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 489–494.
117. Kim, D.; Choi, J.; Yoo, H.; Yang, U.; Sohn, K. Rear obstacle detection system with fisheye stereo camera using HCT. *Expert Syst. Appl.* **2015**, *42*, 6295–6305. [[CrossRef](#)]
118. Ball, D.; Ross, P.; English, A.; Milani, P.; Richards, D.; Bate, A.; Upcroft, B.; Wyeth, G.; Corke, P. Farm Workers of the Future: Vision-Based Robotics for Broad-Acre Agriculture. *IEEE Robot. Autom. Mag.* **2017**, *24*, 97–107. [[CrossRef](#)]
119. Salhi, M.S.; Amiri, H. Design on FPGA of an obstacle detection module over stereo image for robotic learning. *Indian J. Eng.* **2022**, *19*, 72–84.
120. Huang, H.-C.; Hsieh, C.-T.; Yeh, C.-H. An indoor obstacle detection system using depth information and region growth. *Sensors* **2015**, *15*, 27116–27141. [[CrossRef](#)] [[PubMed](#)]
121. Muhovič, J.; Bovcon, B.; Kristan, M.; Perš, J. Obstacle tracking for unmanned surface vessels using 3-D point cloud. *IEEE J. Ocean. Eng.* **2019**, *45*, 786–798. [[CrossRef](#)]
122. Murmu, N.; Nandi, D. Lane and Obstacle Detection System Based on Single Camera-Based Stereo Vision System. In *Applications of Advanced Computing in Systems*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 259–266.
123. Sun, T.; Pan, W.; Wang, Y.; Liu, Y. Region of Interest Constrained Negative Obstacle Detection and Tracking With a Stereo Camera. *IEEE Sens. J.* **2022**, *22*, 3616–3625. [[CrossRef](#)]
124. Dairi, A.; Harrou, F.; Senouci, M.; Sun, Y. Unsupervised obstacle detection in driving environments using deep-learning-based stereovision. *Robot. Auton. Syst.* **2018**, *100*, 287–301. [[CrossRef](#)]
125. Zhang, J.; Hu, S.; Shi, H. Deep learning based object distance measurement method for binocular stereo vision blind area. *Methods* **2018**, *9*, 606–613. [[CrossRef](#)]
126. Zhang, Y.; Song, J.; Ding, Y.; Yuan, Y.; Wei, H.-L. FSD-BRIEF: A Distorted BRIEF Descriptor for Fisheye Image Based on Spherical Perspective Model. *Sensors* **2021**, *21*, 1839. [[CrossRef](#)]
127. Choe, J.; Joo, K.; Rameau, F.; Kweon, I.S. Stereo object matching network. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xian, China, 30 May–5 June 2021; pp. 12918–12924.
128. Luo, W.; Schwing, A.G.; Urtasun, R. Efficient deep learning for stereo matching. In *Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 30 June 2016*; pp. 5695–5703.
129. Song, W.; Yang, Y.; Fu, M.; Li, Y.; Wang, M. Lane detection and classification for forward collision warning system based on stereo vision. *IEEE Sens. J.* **2018**, *18*, 5151–5163. [[CrossRef](#)]
130. Haris, M.; Hou, J. Obstacle Detection and Safely Navigate the Autonomous Vehicle from Unexpected Obstacles on the Driving Lane. *Sensors* **2020**, *20*, 4719. [[CrossRef](#)]
131. Mukherjee, A.; Adarsh, S.; Ramachandran, K. ROS-Based Pedestrian Detection and Distance Estimation Algorithm Using Stereo Vision, Leddar and CNN. In *Intelligent System Design*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 117–127.
132. Tijmons, S.; de Croon, G.C.; Remes, B.D.; De Wagter, C.; Mulder, M. Obstacle avoidance strategy using onboard stereo vision on a flapping wing mav. *IEEE Trans. Robot.* **2017**, *33*, 858–874. [[CrossRef](#)]
133. Lin, J.; Zhu, H.; Alonso-Mora, J. Robust vision-based obstacle avoidance for micro aerial vehicles in dynamic environments. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 2682–2688.
134. Grinberg, M.; Ruf, B. UAV Use Case: Real-Time Obstacle Avoidance System for Unmanned Aerial Vehicles Based on Stereo Vision. In *Towards Ubiquitous Low-Power Image Processing Platforms*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 139–149.
135. Rateke, T.; von Wangenheim, A. Passive vision road obstacle detection: A literature mapping. *Int. J. Comput. Appl.* **2020**, *44*, 376–395. [[CrossRef](#)]
136. Wang, W.; Wang, S.; Guo, Y.; Zhao, Y. Obstacle detection method of unmanned electric locomotive in coal mine based on YOLOv3-4L. *J. Electron. Imaging* **2022**, *31*, 023032. [[CrossRef](#)]