# Ground or Obstacles? Detecting Clear Paths in Vehicle Navigation

Ralph Aeschimann[1], Paulo Vinicius Koerich Borges[2]

***Abstract*— Efficient estimation of obstacles is a crucial component in autonomous vehicle navigation. Fusing stereo vision and laser scan data is a widely used approach for obstacle detection and navigation. Stereo vision can detect obstacles and the ground. Planar laser range finders can detect obstacles, but false positives occur when the scans collide with the ground. This paper proposes an algorithm for cooperative fusion of stereo vision and laser data. The algorithm checks the consensus of the two signals before filtering out the false positive laser scans from ground hits. Experimental results for tests in real driving situations in an industrial environment are presented. The results confirm the effectiveness of the proposed algorithm for robust and safe navigation.**

***Index Terms*— Sensor Fusion, Laser Range Finder, Stereo Vision, Obstacle Detection**

## I. INTRODUCTION

In the context of autonomous driving, perceiving and understanding the environment surrounding a vehicle is essential. Given that all sensors have their advantages and drawbacks, a single sensor is often not sufficient to accurately represent the environment and the use of multiple sensors on vehicles has become a common alternative. Laser range finders and stereo vision are an often used combination as the sensor modalities complement each other: 2D laser range finders allow high precision and high frequency measurements, but they only provide information about the environment in the scanning plane. On the other hand, stereo vision provides dense 3D structural data and texture information. Its drawbacks are the limited field of view, the limited reliable range and the low frequency. In particular, the precision and noisiness of range measurements vary with the lightning conditions and the texture of the environment.

In this paper, a method for filtering out false positive obstacles from laser scans colliding with the ground is presented, using a laser-vision fusion (LVF) framework. The challenge of cooperative fusion is tackled using stereo vision and planar laser range finders. The stereo vision data is processed into a digital elevation model (DEM) which is then used to estimate a ground plane.

### A. Related Work

A significant amount of literature can be found on sensor fusion of laser and stereo vision data. The fusion applications reach fields such as obstacle avoidance, path planning, localization, mapping and SLAM. In early works, the methods were not cooperative [1]. The visual data was simply utilized to validate the correctness of the laser range measurements. Moghadam *et al.* [2] apply sensor fusion for path planning and mapping. The fusion is cooperative, but only on a distinct level: for each sensor a costmap is generated and these costmaps are fused into an occupancy grid [3]. The method works in real-time in complex and cluttered indoor and outdoor environments. Haberjahn *et al.* [4] propose a multi-level fusion framework in which the sensor data is merged on three levels: on low-level using points, on mid-level using objects and on high-level using tracks. The framework was tested for laser range and stereo vision sensors, but the method is independent of the number and type of sensors, as long as the data and error models can be provided. Conflicting sensor statements are solved with competing object information in order to get a more accurate object determination. The method also overcomes the methodology of using one sensor as main sensor determining the object hypotheses and another sensor only for verification [1], [5].

Dense 3D point clouds from stereo vision are generally computationally heavy to process. A convenient approach is to downsample the stereo points into a digital elevation model (DEM). Oniga *et al.* [6] use a DEM to derive a quadratic ground model. The ground model considers the 3D uncertainty from stereo and is refined using a region growing process. Density-based and road surface-based algorithms detect obstacles. The complete framework allows

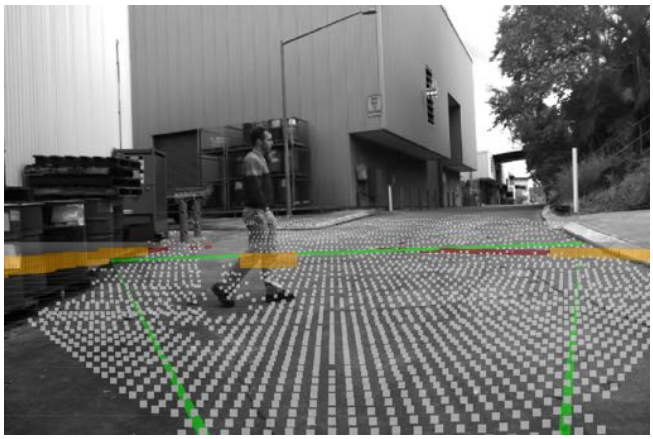


Fig. 1. View of the proposed laser-vision fusion algorithm. The ground plane (green polygon) is estimated from ground points (white). The red points represent the filtered out laser readings, which are false positive obstacles, and the true obstacles are represented by the orange points.

This work was supported by the CSIRO Computational Informatics and by the CSIRO Future Manufacturing Flagship.

[1]Ralph Aeschimann is with the Autonomous Systems Lab at ETH Zurich, Zurich, Switzerland, aralph@student.ethz.ch

[2]Paulo Vinicius Koerich Borges is with the Autonomous Systems Lab at CSIRO Computational Informatics, Brisbane, Australia. Paulo Borges is also an Adjunct Senior Lecturer at the School of Information Technology and Electrical Engineering, University of Queensland, vini@ieee.org

to discriminate between roads, traffic isles and obstacles.

Although the ground usually presents some degree of curvature, a linear model for the ground plane estimation can be assumed. This is a simplification commonly used in literature [7], [8] and has the advantage of lowering the complexity of the algorithm and decreasing its computational costs.

A specific challenge in fusing laser and visual data is to detect laser scans colliding with the ground. If this is not done thoroughly, false positive obstacles remain on the vehicle's path which hinders efficient navigation. In practice, false positives can occur when laser scans hit the ground due to the pitching of the vehicle or the sloped ground geometry as illustrated in Figure 2. In that case, the laser points should not be considered as obstacles. Using competitive sensor information, a fusion framework is able to solve contradictory statements in obstacle detection.

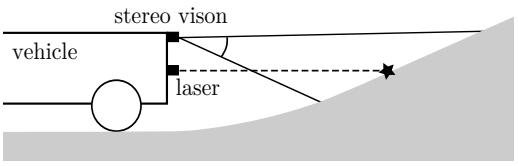



Fig. 2. False positive obstacle detection from laser scans at slopes.

In the context of the DARPA Urban Challenge [9], a framework for an autonomous urban vehicle is presented [10] using planar laser range finders and a Velodyne sensor. One of the challenges is distinguishing obstacles and ground. Using a polar grid around the vehicle, the ground height of each cell is calculated from the infalling Velodyne laser points. These cells are subject to a ground-slope constraint in order to avoid elevated obstacles to be detected as ground. Obstacles are defined as laser points lying a certain threshold above the ground and need to be confirmed by multiple laser hits from one or more laser range finders. The procedure for tuning this threshold is not elaborated. Labayrade *et al.* [11] set up an obstacle detection framework in the automotive context by fusing laser and stereo vision data. The authors use a cooperative fusion approach which takes into account the complementary features of both sensors. The set of obstacles from vision and laser scans are matched. Width and depth of obstacles are provided by laser measurements, whereas the stereo vision provides the height. Several cooperative fusion schemes of stereo vision and laser range finders are discussed and one is implemented and tested. The vehicle pitch, estimated by stereo vision, is used to filter laser scan data such that the laser points colliding with the ground are removed. Unfortunately, this filtering step is not discussed in detail.

As examined by the authors of a multi-level fusion framework [4], low-level fusion contributes greatly to the accuracy of object detection. Accordingly, refining the low-level fusion techniques impacts the whole obstacle detection process the most.

### B. Contribution of this Work

In this paper, a method for filtering out false positive obstacles from laser scans colliding with the ground is presented. Scans from planar laser range finders are combined with data from stereo vision to form the LVF algorithm. In our implementation, the lasers and cameras are mounted on the front of an autonomous electric vehicle, pointing forward. The stereo vision data is processed into a DEM which is then used to estimate a ground plane. The presented method checks the reliability of the laser and stereo vision data by evaluating the consensus of the signals. The method allows for accurate and fast distinction between actual obstacles and false positives assisting towards safe navigation of autonomous vehicles. The method works in real-time in complex and cluttered outdoor environments. In addition to the implementation, this contribution also presents an analysis of the proposed consensus metric in the presence of noise. The analysis provides deeper insight into the signal properties and motivates the tuning of the algorithm parameters.

The paper is organized as follows. Section II presents the algorithm of the ground plane estimation using stereo vision data as input. Section III presents the proposed sensor fusion algorithm explaining the sensor consensus validation and the false positives filtering. Section IV presents a statistical analysis of the signal used in the sensor fusion algorithm. Section V presents practical considerations, experiments and results of the fusion scheme. Finally, conclusions and future work are discussed in Section VI.

## II. Ground Plane Estimation

As a first step to the LVF, an estimate of the ground is necessary. The point cloud from stereo vision is split into ground and non-ground points. The ground points are used to estimate a ground plane.

The coordinate system used in this section has its origin laterally centered in front of the vehicle. The $x$-axis is pointing left, the $y$-axis is pointing up and the $z$-axis is pointing towards the front in the driving direction, as illustrated in Figure 4.

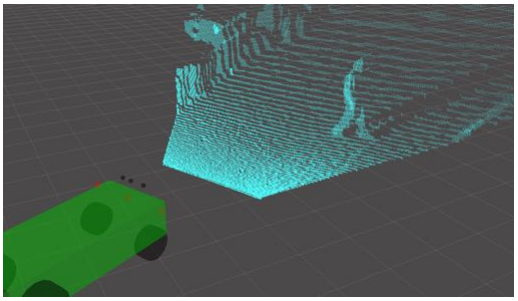
### A. DEM and Ground Points

A dense 3D point cloud from stereo vision is taken as input and downsampled by building a rectangular DEM. The height $h_{\text{cell}}$ of a DEM cell is defined by the maximum height of all points lying within the cell.

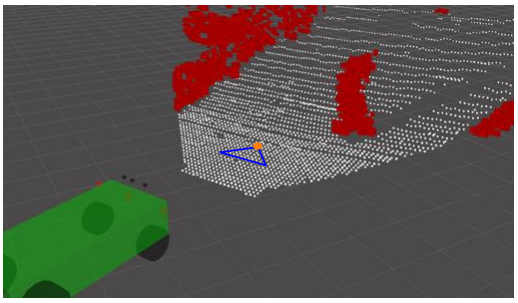
From the DEM cells, we extract the subgroup of ground cells. The first $n_{\text{initial}}$ DEM rows in front of the robot are initialized as ground cells. The remaining cells are tested according to the following criteria:

- a DEM cell is compared with the cells in a search triangle (see Figure 3).
- For each of those cells, the slope gradient $\gamma$ is calculated.
- If the majority of the slope gradients lies below a gradient threshold $\lambda_\gamma$, the cell is accepted as ground.

The threshold $\lambda_\gamma$ and the size of the search triangle must be selected such that the ground points are correctly detected in case of a tilted vehicle, inclined roads and sensor noise. An example of the raw 3D data from stereo vision and the resulting DEM cells is shown in Figure 3.



(a) Unprocessed 3D point cloud from stereo vision (blue)



(b) DEM Ground cells (white) and obstacles (red)

Fig. 3. (a): The 3D points from stereo vision for a typical scene of a road with a pedestrian and obstacles at the side of the road. (b): After building a DEM and applying ground and obstacle detection. The search triangle (blue) for a DEM ground cell (orange).

### B. Ground Plane Coefficients

The ground DEM cells are fed as input points to the ground plane estimation algorithm based on the random sampling consensus (RANSAC) method [12]. The algorithm specifies an upper limit $\alpha_{\max}$ for the angle between the $y$-axis and the output plane normal in order to avoid detecting walls as solution. The output of the algorithm are the coefficients $a_\Sigma, b_\Sigma, c_\Sigma, d_\Sigma$ of the ground plane equation

$$a_\Sigma x + b_\Sigma y + c_\Sigma z + d_\Sigma = 0 \tag{1}$$

An upper limit $d_{\Sigma,\max}$ for the fourth ground plane coefficient is set for the plane estimation to avoid having elevated horizontal planes as a solution.

## III. Sensor Fusion

In the proposed LVF algorithm, a selection of laser points and the ground plane estimate are tested for consensus. Given the consensus, the filtering of the selected laser points is applied.

### A. Selection of Laser Points Relevant for Filtering

In order to navigate on clear paths, the LVF is applied when laser points hit the ground in front of the vehicle. The front is defined by the field of view of the stereo sensor. Laser points lying outside of the front do not need to be included into the filtering process as they do not lie in the driving direction and the ground plane estimation is not available at their positions. Laser points hit the ground due to the dynamic pitching of the vehicle or due to the geometry of the road (slopes, bends or dents) as illustrated in Figure 2.

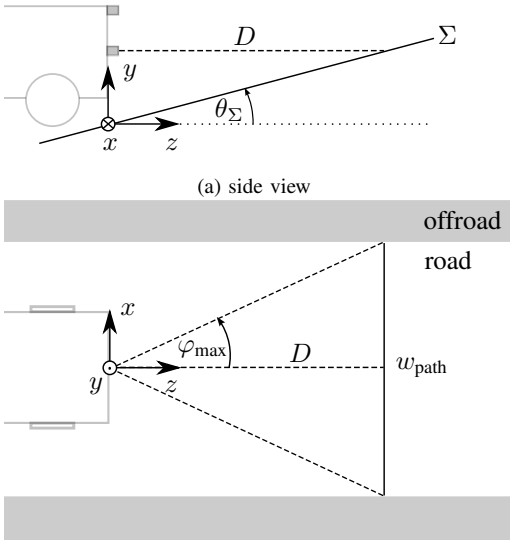
All laser points lie on a plane. The position $\mathbf{p}(i)$ of a laser point with spatial index $i$ is described by its polar coordinates which are angle $\varphi(i)$ and distance $\rho(i)$ from the source of the laser range finder to the point of impact.

We set a maximum allowed angle $\varphi_{\max}$ to select the relevant laser points. $\varphi_{\max}$ depends on the parameter $w_{\text{path}}$, which is the width of the path to be considered for filtering. The maximum allowed angle $\varphi_{\max}$ is approximated by

$$\varphi_{\max} = \arctan\left(\frac{w_{\text{path}}}{2D}\right) \tag{2}$$

$$D = \frac{\cos(\theta_\Sigma) h_{\text{laser}} - d_\Sigma}{\sin(\theta_\Sigma)} \tag{3}$$

where $D$ is the horizontal distance from the laser range finder to the ground, $\theta_\Sigma = \cot^{-1}(-b_\Sigma/c_\Sigma)$ is the pitch angle of the ground plane estimation and $h_{\text{laser}}$ is the mounting height of the laser range finder. The geometry leading to (2) and (3) is visualized in Figure 4 for the case $d_\Sigma = 0$.




(a) side view

(b) top view

Fig. 4. The estimated ground plane $\Sigma$ (solid line), impinging laser scans (dashed lines) and the width of the path $w_{\text{path}}$ (solid line) considered for filtering.

### B. Sensor Consensus

For each laser point classified as relevant, the signed Euclidean distance $d(i)$ to the estimated ground plane $\Sigma$ is calculated. The distance is positive if the laser point lies between the laser sensor source and the plane. For a point at position $\mathbf{p}(i) = (p_x(i), p_y(i), p_z(i))$, the distance $d(i)$ is

defined as

$$d(i) = \frac{a_\Sigma p_x(i) + b_\Sigma p_y(i) + c_\Sigma p_z(i) + d_\Sigma}{\sqrt{a_\Sigma^2 + b_\Sigma^2 + c_\Sigma^2}} \quad (4)$$

The consensus metric $M$ is

$$M = \frac{1}{N} \sum_{i=1}^{N} d(i) \quad (5)$$

where $N$ is the number of laser points considered relevant as explained in Section III-A.

The measurements of $d(i)$ are prone to noise from both sensors and the ground plane estimate is not necessarily precise. The correctness of the measurements is validated by checking the sensor consensus which is given if $M < \lambda_M$, where $\lambda_M$ is the metric threhsold.

### C. Laser Points Filtering

In case of sensor consensus, the laser points are filtered in order to remove the collision points with the ground. The filter is only applied to the laser points for which $d(i) < \lambda_d$. The parameter $\lambda_d$ is the threshold for the distance $d$ applied in the filtering step, whereas $\lambda_M$ is the threshold for the metric $M$ applied in the sensor consensus step. Note that $d(i)$ is a signed distance and therefore all laser points with negative distances (lying on the side of the plane away from the laser) are discarded.

Filtering out obstacle points allows detecting clear paths, but also affects the reliability of the vehicle navigation. In particular, filtering out obstacles close to the vehicle in the driving direction can be hazardous. So far, the minimal distance $D_{\text{min}}$ between a filtered obstacle and the origin of the laser range finder is limited by the parameters $\alpha_{\text{max}}$ (the allowed tilting angle of the estimated ground plane) and $d_{\Sigma,\text{max}}$ (the maximal fourth ground plane parameter). The relation between these parameters is expressed according to

$$D_{\text{min}} = \frac{\cos(\alpha_{\text{max}}) h_{\text{laser}} - d_{\Sigma,\text{max}} - \lambda_d}{\sin(\alpha_{\text{max}})} \quad (6)$$

This formula was derived in the same manner as (3). If a laser point lies closer than a safety distance $D_{\text{safety}}$ to the vehicle, the point is not filtered out by the LVF algorithm. To be able to brake before hitting a static obstacle, the distance $D_{\text{safety}}$ has to be chosen larger than the total stopping distance $D_{\text{stop}}$. An expression of $D_{\text{stop}}$ can be found using the law of conservation of energy [13]. The vehicle's kinetic energy is set equal to the work done by braking which leads to

$$D_{\text{stop}} = v_{\text{veh}} t_{\text{react}} + \frac{v_{\text{veh}}^2}{2 \mu_{\text{fric}} g} \quad (7)$$

where $v_{\text{veh}}$ is the vehicle's velocity at the the start of braking, $t_{\text{react}}$ is the perception-reaction time, $\mu_{\text{fric}}$ is the coefficient of friction between the road surface and the tires and $g$ is the acceleration of gravity. In the experiments in this paper, a formula of type $D_{\text{stop}} = a v_{\text{veh}} + b v_{\text{veh}}^2$ with empirical parameters $a, b$ has been applied to calculate the total stopping distance.

## IV. Statistical Signal Analysis

In Section III, we discussed how the distance metric $d$ from a laser point to the estimated ground plane is calculated. In this section, a statistical analysis on this signal is presented providing a better understanding of the metric. The theoretical probability of a false positive obstacle detection is derived aiding in the process of tuning the parameters of the sensor consensus and the laser point filtering.

The notation $\mu_z$ stands for the temporal statistical mean and $\sigma_z^2$ stands for the temporal statistical variance of a function $z(i,t)$ with spatial index $i = 1...N$ at time $t$. The expected values $\text{E}[.]$ discussed in the following are with respect to $t$.

### A. Measurements Model

We model the measured distance $d(i,t)$ as composed of a constant term $c$ representing the "noise-free" distance and a zero-mean additive white Gaussian noise (AWGN) term $\eta(i,t)$. In practice, $\eta(i,t)$ is correlated over space and this correlation is modeled by a low-pass filter $h(i)$. The autocorrelation function of the signal is illustrated in Figure 6. Based on these assumptions, $d(i,t)$ is written as

$$d(i,t) = c + \eta_h(i,t) \quad (8)$$

with $\eta_h(i,t) = h(i) * \eta(i,t)$. Because $\mu_\eta = 0$, $\mu_d(i) = \mu_c$. In addition, $\sigma_d(i) = \sigma_{\eta_h}(i)$.

### B. Statistics of the Metric

Inserting (8) into (5), the theoretical mean value of the metric $M$ is

$$\mu_M = \text{E}[M(t)] = \frac{1}{N} \sum_{i=1}^{N} (\text{E}[c] + \text{E}[\eta_h(i,t)]) \quad (9)$$

Because $\eta$ is zero-mean, $E[\eta_h(i,t)] = 0$. Therefore,

$$\mu_M = \frac{1}{N} \sum_{i=1}^{N} \text{E}[c] = c \quad (10)$$

Ideally, $c = 0$ on an obstacle-free road. In practice, however, the precision of the sensors is limited, the transformation estimate between the stereo cameras and the laser range finder is not exact and the ground plane estimation is only an approximation of the real ground. The theoretical variance of the metric is

$$\sigma_M^2 = \text{E}[M(t)^2] - \mu_M^2 \quad (11)$$

Inserting (10) into (11) and (8) into (5) leads to

$$\begin{aligned} \sigma_M^2 &= \text{E}\left[ \left( \frac{1}{N} \sum_{i=1}^{N} (c + \eta_h(i,t)) \right)^2 \right] - c^2 \\ &= \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \Big( \text{E}[c^2] + 2\text{E}[c\,\eta_h(j,t)] \\ &\quad + \text{E}[\eta_h(i,t)\,\eta_h(j,t)] \Big) - c^2 \end{aligned} \quad (12)$$

Since $\eta_h(i)$ is zero-mean and is not correlated to $c$, the expected value $\mathrm{E}[c\,\eta_h(j,t)] = 0$. Rewriting,

$$\sigma_M^2 = c^2 + \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\mathrm{E}[\eta_h(i,t)\eta_h(j,t)] - c^2$$

$$\sigma_M^2 = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\mathrm{E}[\eta_h(i,t)\eta_h(j,t)] \quad (13)$$

Given that the autocorrelation $r_x$ of a signal $x$ in random processes is defined as $r_x = \mathrm{E}[x(i)x(j)]$ [14], (13) can be written as

$$\sigma_M^2 = \frac{1}{N}\sum_{j=1}^{N} r_{\eta_h}(j) \quad (14)$$

In the Appendix, we show that $r_{\eta_h}(j) = \sigma_\eta^2 r_h(j)$. Therefore, the variance of $M$ can be written as

$$\sigma_M^2 = \frac{1}{N}\sum_{j=1}^{N}\sigma_\eta^2 r_h(j) = \frac{\sigma_\eta^2}{N}\sum_{j=1}^{N} r_h(j) \quad (15)$$

where $r_h$ is the autocorrelation function of the impulse response of filter $h$.

### C. Probabilities of False Positive Obstacle Detections

Once $\mu_M$ and $\sigma_M^2$ are determined, it is possible to estimate the probability of a false positive obstacle detection based on the model assumptions. This is particularly relevant to estimate how the threshold of metric $M$ impacts the performance of the algorithm. For this task, we employ the Q-function $Q(\lambda_x)$, the complement of the cumulative distribution function $\Phi(\lambda_x)$ for a Gaussian random variable $x$ with mean $\mu_x$ and variance $\sigma_x^2$. The Q-function is an expression for the probability that the Gaussian random variable $x$ will obtain a value larger than $\lambda_x$. It is defined as [15]

$$Q(\lambda_x) = 1 - \Phi(\lambda_x) = \frac{1}{2}\,\mathrm{erfc}\left(\frac{\lambda_x - \mu_x}{\sqrt{2\sigma_x^2}}\right) \quad (16)$$

where erfc is the complementary error function [16]. Using the Q-function, an expression for the theoretical probability of a false positive obstacle detection at a distance threshold $\lambda_M$ is derived. In other words, this probability is the rate at which the metric $M$ exceeds a threshold $\lambda_M$. This probability is denoted as $P_{\text{theo}}(\lambda_M)$ and is defined as

$$P_{\text{theo}}(\lambda_M) = \frac{1}{2}\,\mathrm{erfc}\left(\frac{\lambda_M - \mu_M}{\sqrt{2\sigma_M^2}}\right) \quad (17)$$

To validate this formula, we present simulation and real experimental results in the next section.

## V. EXPERIMENTS

In this section, the results of the experimental evaluation of the fusion algorithm and the statistical signal analysis are presented.

### A. Practical Considerations

The algorithm is implemented in C++ and runs on the Robotics Operating System (ROS) [17]. The Point Cloud Library [18] was used to implement the RANSAC ground plane estimation.

For the lasers, two Hokuyo range sensors were used. The sensors were placed on the left and on the right in the front of the vehicle, both mounted at height $h_{\text{laser}} \approx 0.79$ m. The signals of the laser scanners were fused to create a single laser scan signal with a sensor origin centered in the front of the vehicle as depicted in Figure 4. For this virtual sensor origin, the derived formulae for $\varphi_{\max}$ and $D$ in Section III represent good approximations. The laser scanners have an angular resolution of $0.25°$ and a scan angle of $270°$. The laser scanners were running at a frequency of 30 Hz and have a maximum detection distance of 30 m. The stereo vision sensor was composed of two monochromatic Point Grey Grasshopper cameras holding a $2/3$" CCD imaging sensor fitted with Kowa lenses of 5 mm focal length and aperture $F1.8$. The resolution was set to $800 \times 600$ pixels running at a frame rate of 15 Hz. The system was mounted at a height of 1.0 m on a rig with a baseline of 25 cm as illustrated in Figure 5. The sensors were mounted on a John Deere Gator, an electric medium size utility vehicle (see Figure 5), and driven in unstructured and structured industrial environments. The

Fig. 5. The test vehicle, a John Deere Gator holding multiple sensors of which the two laser range finders (red boxes) and the stereo vision sensor (blue box) were used.

computations were executed on an Intel Core $i7$ quadcore processor running at 2.7 GHz. The average computational time was 20 ms for the DEM and the ground point extraction, less than 1 ms for the ground plane estimation and less than 1 ms for the sensor fusion. The sensor fusion node operated at 15 Hz.

In our implementation, the upper limit for the angle between the $y$-axis and the plane normal is $\alpha_{\max} = 0.52$ rad, the upper limit for the fourth ground plane coefficient is $d_{\Sigma,\max} = 0.6$ m, the width of the path is $w_{\text{path}} = 5$ m and the parameters for the total stopping distance equation are $a = 1.2$ s and $b = 0.25\,\text{s}^2/\text{m}$. The filter $h$ was designed as finite impulse response (FIR) filter approximated by

$$h(i) = \frac{1}{11}\sum_{k=0}^{10}\delta(i-k) \quad (18)$$

The autocorrelation over space of the signal $d(i)$ is shown in Figure 6. The low-pass characteristic of this signal has been handled by using a low-pass filtered noise model.

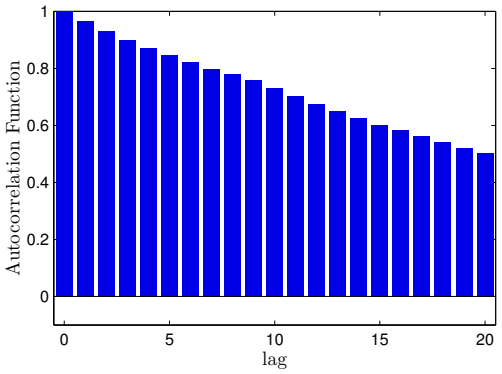



Fig. 6. The sample autocorrelation function of the signal $d(i)$ with spatial index $i$.

### B. Threshold Experiments

An obstacle-free road was used as a testing environment for an initial validation of the statistical signal analysis. The vehicle was driven in areas where the laser scans were constantly hitting the ground. Extensive test data has been gathered for this scenario and the probabilities $P_{\text{theo}}$ of $M$ exceeding a threshold $\lambda_M$ have been calculated. The results are illustrated in Figure 7. The theoretical curve is a plot of (17). The simulation results were generated using a random variable with mean $\mu_M$ and variance $\sigma_M^2$. The overlay of the theoretical curve (depicted as solid line) and the simulation results (represented by crosses) illustrates the validity of the statistical signal analysis. The experiment data (represented by circles) does not correspond to the simulation data for larger thresholds. This discrepancy occurs since the probabilities of $M$ exceeding large thresholds are small. For those small probabilities, the experimetally determined values do not approximate the expected values well.

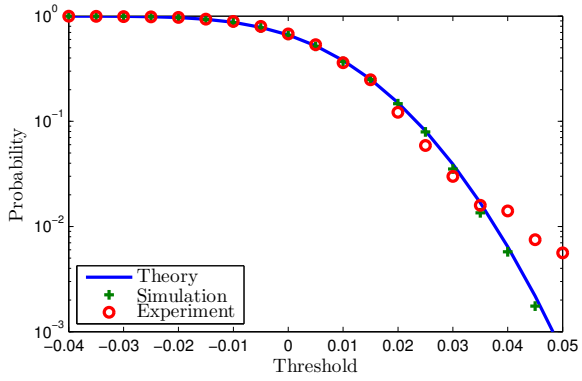



Fig. 7. Probabilities of $M$ exceeding a threshold as a function of the threshold. Displayed are the curve from the theoretical analysis, a simulation and an experiment based on real data.

The tuning of the thresholds of the laser filter algorithm was based on datasets with obstacles on the road. The metric threshold was set to $\lambda_M = 0.35\,\text{m}$ and the distance threshold was set to $\lambda_d = 0.20\,\text{m}$.

### C. Closed-Loop Implementation on an Autonomous Vehicle

The sensor fusion system was implemented on an autonomous vehicle to allow closed-loop test runs. These tests display the ability and effectiveness of the algorithm to assist in autonomous navigation. The Gator was used as the test platform, with the sensor setup and parameter settings as described in Section V-A. The vehicle has been automated by our group at CSIRO[1] and the tests have been run on our site at QCAT[2], a large industrial park in Australia. The navigation of the vehicle is done by the ROS Navigation Stack [17] which can load a predefined laser-generated map and can localize the vehicle in this map using the laser scanners. The navigation stack runs a costmap which is constantly updated with the obstacles detected by the laser range finders. The costmap is the basis for planning paths.

The performance of the algorithm is evaluated in different areas. In all of them, the vehicle is given a goal approximately 50 m away from its initial position. The vehicle plans a path and starts to drive autonomously along a road towards the goal. On the road to the goal, the vehicle reaches a slope which causes the laser scans to hit the ground in front of the vehicle, as illustrated in Figure 2. We consider three scenarios, which evaluate *1)* the case of not using any laser scan filtering, *2)* the performance of the laser scan filtering in case of an obstacle-free road and *3)* the performance of the laser scan filtering in case of obstacles on the road.

*1) No Laser Filtering:* This scenario covers the case where the proposed laser filtering algorithm is *not* applied. This situation is illustrated in Figure 8. When the vehicle reaches the slope, the laser scans hit the ground. This leads to the costmap containing false positive obstacles which interfere with the path planning of the vehicle. The vehicle tries to replan the path, but finds no solution to follow the road without hitting the false positive obstacles, and the vehicle is stalled. 25 tests in different locations were run in this scenario. In all of them, the vehicle was to drive past the slope and complete the mission.

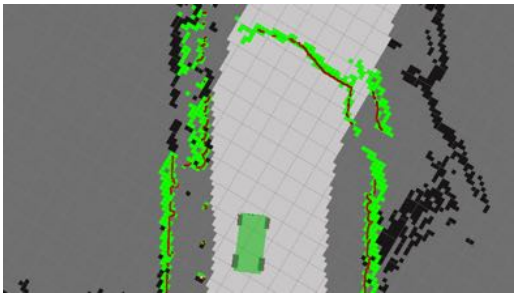
*2) Laser Filtering on Obstacle-free Road:* We consider the same environment as above, but in this scenario the laser filtering module is active. As the vehicle approaches the slope, the laser filtering algorithm removes laser scans before passing them to the navigation module.

For this scenario, a total of 25 runs on different locations have been made. In all of them, the vehicle reached the goal. The false positive obstacles of the previous scenario that stalled the navigation are now successfully removed by the laser filter.

*3) Laser Filtering with Obstacles on the Road:* This scenario contains an obstacle on the sloped road which the vehicle has to pass to reach the goal. The vehicle cannot sense the obstacle at the beginning of the run and plans a

[1]The Commonwealth Scientific and Industrial Research Organisation.
[2]Queensland Centre for Advanced Technologies.

(a) Left camera view



(b) Top-down costmap view

Fig. 8. Navigation failure due to false positive obstacle detection from the laser scans (red) hitting the ground. The costmap contains obstacles (green) which are false positives hindering path planning through the clear road.

path through the location of the obstacle. While approaching the slope, the laser scans hit both the ground (false positive obstacle) and the actual obstacle (true positive obstacle). In order to reach the goal, the vehicle must filter out the false positive obstacles while keeping the real obstacle.

The performance of the algorithm was evaluated by measuring (i) how often the obstacle on the road was detected, (ii) how often a collision with the obstacle was avoided by either driving around it or stopping before hitting it, as well as (iii) how often the goal was reached. The used obstacles were cones, poles as well as static and moving pedestrians. The height of the obstacles lay between $0.3\,\mathrm{m}$ and $1.8\,\mathrm{m}$. Roads with different ground geometries were used for the test runs. The results of this scenario are shown in Table I, which displays the the three performance criteria mentioned above. Snapshots of the running algorithm are given in Figures 1 and 9, with both images displaying different views of the same scene.

TABLE I

PERFORMANCE RESULTS FOR THE LASER FILTERING IN A SCENARIO WITH OBSTACLES.

| | |
|---|---|
| Number of Runs | 25 |
| Detected true positive | 96% |
| Avoided true positive | 96% |
| Goal reached | 92% |

In one test, the algorithm failed to detect and avoid the obstacle. In this run, the road had a strong curvature and the obstacle was a cone of height $0.3\,\mathrm{m}$. The low height of the obstacle, its position and the imprecision of the ground estimate caused the obstacle to be filtered out.

The laser filtering allows for false positives obstacles to be filtered out from noisy measurements, but it can also remove true positives if their height lies below the threshold $\lambda_d$. Therefore, the obstacles used in the experiment had heights larger than this threshold. As mentioned, removal of true positive obstacles with height larger than $\lambda_d$ can occur, but becomes less probable with increasing height.

The vehicle did not reach the goal in two different runs. In one run, the vehicle stopped close to the obstacle which caused the replaning of the path to fail. This failure is not related to the performance of the laser filtering algorithm. In the other run which did not reach the goal, the reason for failure was the incomplete removal of false positive obstacles. After filtering, false negative obstacles remained at the side of the road. The cleared path was too narrow for the vehicle to pass which caused the vehicle to stop in front of the false positive obstacles. The amount of removed false positives is related to the accuracy of the ground plane estimate representing the ground.

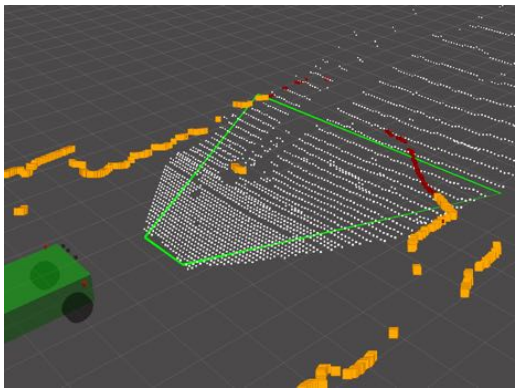


Fig. 9. The laser filtering sensor fusion algorithm. Estimated ground plane (green polygon) from ground points (white). The filtered out laser points (red) which are false positive obstacles and the remaining points (orange).

## VI. CONCLUSIONS AND FUTURE WORK

A method for handling laser scans hitting the ground has been presented. The method is simple and efficient in computational resources. The method was implemented and tested on an autonomous vehicle on closed-loop experiments. The results show that thanks to the laser filtering, clear paths could be detected. The tuning of threshold parameters was motivated by a statistical signal analysis. However, the performance is limited by the accuracy of inter- and intrasensor calibration and by the used models, such as the ground plane estimation.

In future work, the ground plane model could be replaced by either a quadratic ground model or by a set of planar ground segments. This would allow the system to perform better on strongly curved roads. Alternatively, the distance between a laser point and its closest DEM ground cell could be used as distance measurement. No model assumption would be requried for this approach, but the DEM ground

point selection might need to be refined. These suggested alternative methods are easy to implement, but may rise the computational costs of the algorithm. Additional future work includes improving the sensor consensus evaluation. This could be done by defining new metrics for the sensor consensus and comparing them with the one used in this work.

## APPENDIX

A linear time-invariant system $\eta_h$ is set up using the convolution summation of a low-pass filter $h$ and unfiltered AWGN $\eta$ according to

$$\eta_h(i,t) = \sum_{k=-\infty}^{\infty} h(k)\eta(i-k,t) \tag{19}$$

For a more compact notation, from here until the end of this section, the temporal index $t$ will be dropped from the notation. Multiplying both sides by the complex conjugate $\eta_h^*(i-l)$ and taking the expectation leads to

$$\begin{aligned}&\mathrm{E}[\eta_h(i)\eta_h^*(i-l)] = \\ &\quad\sum_{k=-\infty}^{\infty} h(k)\mathrm{E}[\eta(i-k)\eta_h^*(i-l)]\end{aligned} \tag{20}$$

The equation is rewritten by using the notation $r_x$ for the autocorrleation of a signal $x$ and $r_{xy}$ for the cross-correlation of signals $x, y$.

$$\begin{aligned} r_{\eta_h}(l) &= \sum_{k=-\infty}^{\infty} h(k) r_{\eta\eta_h}(l-k) \\ &= h(l) * r_{\eta\eta_h}(l) \end{aligned} \tag{21}$$

Using the property $r_{\eta\eta_h}(l) = h^*(-l) * r_\eta(l)$ [14], this can be rewritten to

$$\begin{aligned} r_{\eta_h}(l) &= r_h(l) * r_\eta(l) \\ &= \sum_{m=-\infty}^{\infty} r_h(l-m) r_\eta(m) \end{aligned} \tag{22}$$

The autocorrelation of the noise $\eta$ corresponds to

$$\begin{aligned} r_\eta(m) &= \sum_{k=-\infty}^{\infty} \eta(k)\eta^*(k-m) \\ &= \begin{cases} \sigma_\eta^2, & \text{if } m = 0 \\ 0, & \text{otherwise} \end{cases} \end{aligned} \tag{23}$$

Inserting this into (22) leads to

$$r_{\eta_h}(l) = \sigma_\eta^2 r_h(l) \tag{24}$$

## REFERENCES


[1] H. Baltzakis, A. Argyros, and P. Trahanias, "Fusion of laser and visual data for robot motion planning and collision avoidance," *Machine Vision and Applications*, vol. 15, no. 2, pp. 92–100, 2003. [Online]. Available: http://dx.doi.org/10.1007/s00138-003-0133-2

[2] P. Moghadam, W. Wijesoma, and D. J. Feng, "Improving path planning and mapping based on stereo vision and lidar," in *Control, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference on*, Dec 2008, pp. 384–389.

[3] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol. 22, no. 6, pp. 46–57, June 1989.

[4] M. Haberjahn and K. Kozempel, "Multi level fusion of competitive sensors for automotive environment perception," in *Information Fusion (FUSION), 2013 16th International Conference on*, July 2013, pp. 397–403.

[5] S. Wender and K. Dietmayer, "3d vehicle detection using a laser scanner and a video camera," *Intelligent Transport Systems, IET*, vol. 2, no. 2, pp. 105–112, June 2008.

[6] F. Oniga and S. Nedevschi, "Processing dense stereo data using elevation maps: Road surface, traffic isle, and obstacle detection," *Vehicular Technology, IEEE Transactions on*, vol. 59, no. 3, pp. 1172–1182, March 2010.

[7] S. Nedevschi, R. Danescu, D. Frentiu, T. Marita, F. Oniga, C. Pocol, R. Schmidt, and T. Graf, "High accuracy stereo vision system for far distance obstacle detection," in *IEEE Intelligent Vehicles Symposium*, 2004, pp. 292–297.

[8] J. Zhou and B. Li, "Robust ground plane detection with normalized homography in monocular sequences from a robot platform," in *Image Processing, 2006 IEEE International Conference on*, Oct 2006, pp. 3017–3020.

[9] M. Buehler, K. Iagnemma, and S. Singh, *The DARPA urban challenge: autonomous vehicles in city traffic*. springer, 2009, vol. 56.

[10] J. Leonard, J. How, S. Teller, M. Berger, S. Campbell, G. Fiore, L. Fletcher, E. Frazzoli, A. Huang, S. Karaman, O. Koch, Y. Kuwata, D. Moore, E. Olson, S. Peters, J. Teo, R. Truax, M. Walter, D. Barrett, A. Epstein, K. Maheloni, K. Moyer, T. Jones, R. Buckley, M. Antone, R. Galejs, S. Krishnamurthy, and J. Williams, "A perception-driven autonomous urban vehicle," *Journal of Field Robotics*, vol. 25, no. 10, pp. 727–774, 2008. [Online]. Available: http://dx.doi.org/10.1002/rob.20262

[11] R. Labayrade, C. Royere, D. Gruyer, and D. Aubert, "Cooperative fusion for multi-obstacles detection with use of stereovision and laser scanner," *Autonomous Robots*, vol. 19, no. 2, pp. 117–140, 2005. [Online]. Available: http://dx.doi.org/10.1007/s10514-005-0611-7

[12] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981. [Online]. Available: http://doi.acm.org/10.1145/358669.358692

[13] M. Narasimhan, *Principles of Continuum Mechanics*, ser. Wiley-Interscience publication. Wiley, 1993.

[14] D. G. Manolakis, V. K. Ingle, and S. M. Kogon, *Statistical and adaptive signal processing: spectral estimation, signal modeling, adaptive filtering, and array processing*. Artech House Norwood, 2005, vol. 46.

[15] L. Brennan and I. Reed, "A recursive method of computing the q function (corresp.)," *Information Theory, IEEE Transactions on*, vol. 11, no. 2, pp. 312–313, Apr 1965.

[16] M. Abramowitz and I. A. Stegun, *Handbook of mathematical functions: with formulas, graphs, and mathematical tables*. Courier Dover Publications, 1972, no. 55.

[17] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, no. 3.2, 2009, p. 5.

[18] A. Aldoma, Z.-C. Marton, F. Tombari, W. Wohlkinger, C. Potthast, B. Zeisl, R. B. Rusu, S. Gedikli, and M. Vincze, "Point cloud library," *IEEE Robotics & Automation Magazine*, vol. 1070, no. 9932/12, 2012.