This item is the archived peer-reviewed author-version of:

Fast ground detection for range cameras on road surfaces using a three-step segmentation

Reference:

Van Crombrugge Izaak, Ben Azza Ibrahim, Penne Rudi, Van Barel Gregory, Vanlanduit Steve.- Fast ground detection for range cameras on road surfaces using a three-step segmentation

Lecture notes in computer science - ISSN 0302-9743 - Cham, Springer, 2017, p. 479-490

Full text (Publisher's DOI): http://dx.doi.org/doi:10.1007/978-3-319-70353-4_41

Fast Ground Detection for Range Cameras on Road Surfaces using a Three-Step Segmentation

Izaak Van Crombrugge¹, Ibrahim Ben Azza¹, Rudi Penne^{1,2}, Gregory Van Barel¹, and Steve Vanlanduit¹

¹ Faculty of Applied Engineering, University of Antwerp, Groenenborgerlaan 171, B-2020 Antwerp, Belgium.

{izaak.vancrombrugge, ibrahim.benazza, rudi.penne, gregory.vanbarel, steve.vanlanduit}@uantwerpen.be

² Department of Mathematics, University of Antwerp, Belgium.

Abstract. We present in this paper a fast and simple free floor detection method. Compared to existing methods the proposed method can handle non-planar camera motion by means of a Three-Step procedure. A fast initial segmentation is followed by an intermediate floor plane estimation to adapt to the camera motion. Then, a final segmentation is done using the estimated plane. This allows for correct segmentation even when the camera moves up and down, tilts or rolls. Outdoor measurements of a road surface were performed with a Time-of-Flight camera mounted in front of a car. The measurements contain three types of road surface: concrete, stone and asphalt. The proposed segmentation takes less than 1.25 ms per frame for range images with a resolution of 176 by 132, making it fit for real-time applications. The resulting accuracy is higher than the state of the art.

Keywords: Range camera, Time-of-flight, RGB-D, Floor segmentation, Obstacle detection

1 Introduction

A Time-of-Flight (ToF) camera is a type of range imaging camera that was introduced two decades ago. The camera has several uses such as tracking objects, object scanning, gesture recognition, and measuring volume [1]. Compared to laser range scanners, the ToF technology has two main advantages: it has no moving parts, and a complete image can be taken in a single exposure. This way, frame rates close to 100 frames per second are possible [2]. ToF cameras can deliver grayscale intensity images as well as depth images directly with low processing power.

Although general plane segmentation in range data is a well studied, algorithms optimized for floor detection in ToF images are rare. In the proposed method, the specific geometric properties of a ground plane when seen from a mobile platform are employed to get fast and accurate results. The accuracy of the algorithm is validated by comparing the results of ToF road measurements with a manually constructed ground truth.

1.1 Time-of-Flight Cameras in the Automobile Industry

The ToF technology has a lot of potential in the automotive industry. ToF cameras can be used to gather an accurate measurement of the driver's head position and orientation by analyzing the real-time depth and intensity image of the driver's upper torso and head [3]. It can be used for driver awareness and fatigue detection, and driver distraction detection [4].

Range cameras such as ToF can also be used to monitor the road ahead to measure the road surface, for self-localization, and to detect obstacles [5]. Nowadays, most of the localization and navigation assistance systems depend on the Global Positioning System (GPS) [6]. However, these solutions fall short in tunnels or in underground parking lots due to signal loss. Moreover, GPS cannot provide local information about the obstacles in front of or near the vehicle. Radar is often used for obstacle detection whereas different methods have been used to solve the issue of navigation and self-localization using laser scanners, inertial sensors, vision and sonar. ToF can offer a valuable contribution of dense range information while being durable and robust against vibrations as it is a solid state technology.

1.2 Ground Plane Estimation and Obstacle Detection

The goal of the proposed technique is to detect obstacles in front of a vehicle by the use of a ToF camera looking forward to the road ahead. The obstacles will be detected by segmenting the ground plane. Every point that does not belong to the ground plane is categorized as being part of an obstacle. Several ground plane estimation techniques have been used for 2D cameras using methods such as least squares [7], where Andreas [8] uses stereo depth information to support its ground plane estimation. Probabilistic ground plane estimation methods are presented by Kwon in [9].

Ground plane estimation specifically for 3D range sensors is not widely adopted. Mufti, Faisal and Heinzmann [10] have formulated in their work for Advanced Driver Assistance Systems an algorithm for ground estimation by using 3D video data, captured by the ToF camera, whereas [11] uses a simple method, optimized for floor detection. Ground plane detection using an RGB-D sensor by means of curve fitting is addressed in [12] by Kircali.

More generally, RANdom SAmple Consensus (RANSAC) is often used to fit planes [13]. Also J-linkage could be used to detect planes [14]. An other approach is to employ the Iteratively Reweighted Least Squares (IRLS) for the ground estimation. The ILRS method requires a weight function and a tuning parameter that need more complex heuristics [15].

Each of these methods works iteratively, making either the needed time or the resulting accuracy unpredictable. To reach a predictable accuracy within a fixed time, we use a novel Three-Step method.

1.3 Conditions

In [11] a technique was proposed to segment the free floor surface in range camera data, which is limited to planar motion. It requires that the following conditions are met:

1. The camera is mounted on the mobile platform at a fixed height and angle with respect to the surface.
2. The floor plane is a flat surface.
3. Translations of the camera are parallel to the floor plane.
4. The axis of rotation is always the yaw axis, perpendicular to the floor plane.

In this paper, we present a novel method which loosens these conditions so that the camera angle and height with respect to the floor plane can vary. Condition number 2 remains unchanged, as the ground is modelled as a plane. The suspension travel causes vertical, pitch, and roll movement of the camera. These small movements are allowed in our approach.

2 Proposed Procedure

There are two main parts in the process, as depicted in the flow-chart of Figure 1:

1. The calibration, which can be done offline or in real-time
2. The real-time ground detection

In the first part (calibration), the ground plane is determined by using an accurate and robust plane-fitting method, like the RANSAC plane detection method. The calibration is only performed once in the process, and hence it is acceptable to use a procedure which is relatively time-consuming. The second part is the actual real-time ground detection. This algorithm should be executed in less than 10 ms, as it is performed for every frame.

2.1 Calibration

While driving around with the vehicle, the position and angle of the ground plane relative to the camera varies slightly dependant on the suspension movement. When observed over many frames, the ground plane will be very close to an *average* ground plane most of the time as the vehicle movement can be approximated as planar motion. The knowledge that this average ground plane is invariant for planar motion is used to implement automatic calibration. While driving, the average ground plane can be detected as a constant plane because, relative to the ToF camera, most other planes are moving. This calibration can be done on a previously recorded data set (offline) or based on the previous N frames on the go (real-time).

Fig. 1: Representation of the data flow with its two main parts: offline calibration and real-time ground detection.

The automatic calibration method comprises the following steps:

1. Capture N image frames while moving.
2. Sample K points from every frame,
3. Merge the sampled points of all frames into one point cloud.
4. Extract the plane that contains the most points in the point cloud using a RANSAC plane-fitting algorithm.

In practice, the point cloud would become too large if all points of all frames were used. To reduce the number of points, the amount of captured images N should be chosen large enough to provide enough movement to filter out all non-ground planes. Therefore, K is chosen so that the point cloud has a practical size ($= N \cdot K$ points) for the chosen RANSAC plane fitting algorithm. This practical size must be tailored to the real-time processing unit's memory capacity and processing power. With $N = 1000$ and $K = 20$ the calibration implemented in Matlab takes less than 7 ms on a 2.8 GHz Intel i7 quad core processor.

The result is a *floor plane* π given by the equation:

$$\pi \leftrightarrow ax + by + cz + d = 0 \quad (1)$$

The parameters a, b, c of Equation 1 are normalized so that $a^2 + b^2 + c^2 = 1$. The sign of d is chosen so that the normal vector $(a, b, c) = \vec{n}$ is pointing towards the ToF camera.

In our experiment, all processing was done offline. The calibration was performed on the first 1000 frames of a total number of 3700. Since the calibration was not executed in real-time and considering the low resolution (176×132 pixels) of the IFM O3D303, no subsampling was done.

2.2 Real-Time Ground Detection

The proposed method consists of following three steps:

1. Initial segmentation
2. Ground plane update
3. Final segmentation

To adapt to the varying ground plane, a plane fitting technique is needed. Robust plane segmentation algorithms, like RANSAC, are computationally expensive because they must cope with outliers. To avoid this, we do a first simple segmentation with the calibrated ground plane. The resulting initial segmentation is only partially correct, but it got rid of most of the outliers. The correct plane can be found from the resulting ground plane points with a simple least square fit. Using this plane, a final segmentation is done.

By using three simple steps, the computational complexity stays very limited, resulting in fast calculation times. The novel approach builds further on the method of [11] the so-called *Single-Step Segmentation*. Because the Single-Step Segmentation is the first step in our novel algorithm, we first summarize it here.

Step 1: Initial Segmentation. The calibration results in four scalars that determine the floor plane: a , b , c and d . Knowing that $\|\vec{n}\| = 1$, the signed distance δ from any given point $P = (p_x, p_y, p_z)$ to the floor plane π is given by the following equation:

$$\delta = (p_x \ p_y \ p_z \ 1) \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \quad (2)$$

Since $\vec{n}$ points towards the camera, $\delta > 0$ for points below the floor plane and $\delta < 0$ for points above the floor plane, such as the points of obstacles

The segmentation between floor points and obstacle points is done by comparing the distance δ of every point to the thresholds t_1 and t_2 as shown in Figure 2. All points in the green region between t_1 and t_2 are considered free floor by the initial segmentation step.

Using two thresholds makes the algorithm robust against multipath distance overestimation, a frequently occurring error in time-of-flight measurements [16, 17]. Multipath deformations are located below the expected floor plane and have a positive distance to the plane ($\delta > 0$) of about 100 to 150 mm. Obstacles, on the other hand, are located above the plane and have a negative distance ($\delta < 0$). By choosing $|t_2| > |t_1|$ the multipath error can be ignored while still correctly detecting small obstacles. Best results were obtained with $t_2 = 3 \cdot t_1$.

Fig. 2: Time-of-Flight cameras tend to overestimate the distance near corners due to the multiple paths the reflected light can take.

Fig. 3: Illustration of false positives of the initial ground detection (a) and the final ground detection (b) on indoor measurements using a Kinect V2 ToF camera mounted on a wheelchair.

Step 2: Ground plane update. The initial segmentation, shown in Figure 3(a), has a very low False Positive Rate even in frames with relatively low accuracy. This means that parts of the floor are mislabeled as obstacles but the pixels that are labeled as free floor are reliable. This behaviour can be used to temporarily replace the calibrated floor plane by a better fitting plane. This better plane is obtained by doing a least squares plane-fitting on only the points that were labeled as free floor in the initial segmentation.

Step 3: Final segmentation. The resulting plane is used to do a second segmentation using exactly the same calculation that was used in the first one. The result is what we call the final segmentation as visualized in Figure 3(b).

3 Experiment Description

3.1 Road Measurements

For the experiment, two different Time-of-Flight cameras were used. First, the Kinect V2 ToF camera was mounted on a fixed location in the front of the

Fig. 4: Our instrumented vehicle with two Time-of-Flight cameras: the Microsoft Kinect V2 ToF and the IFM O3D303. Note: the images of the Kinect were not used due to the low quality of the outdoor measurements.

windshield of the vehicle as depicted in Figure 4. Second, the IFM O3D303 camera was mounted under the right headlamp of the vehicle (passenger side), oriented towards the ground surface. This experiment was conducted with the instrumented vehicle³ driven by a human driver. The measurements were carried out in the vicinity of the university campus Groenenborger at driving speed below 20 km/h.

To minimize the influence of ambient light, the experiment was conducted just before sunrise. Nevertheless, the images of the Kinect camera were of very low quality due to many invalid pixels and a high noise level. Therefore, we did not use any data of the Kinect outdoor measurements in the analysis in this paper.

3.2 Indoor Measurements

To test the algorithm on range images with higher resolution, indoor measurements were performed by the use of a Kinect V2 ToF camera (512×424) mounted on a wheelchair. The wheelchair was driven around in the building at a velocity of 5 km/h.

4 Results

4.1 Road Measurements Results

Using the IFM O3D303 Time-of-Flight camera on the instrumented vehicle, measurements were done over 3700 frames. First, the calibration in Section 2.1 was executed on the first 1000 frames. All calculations were done in Matlab.

³ We would like to thank Hyundai BELUX for sponsoring the Hyundai Veloster.

Fig. 5: Accuracy calculated over all 3700 frames in (a), (b) and accuracy calculated for selected frames (c), (d) for both methods. The selected frames are those that had a False Negative Rate > 10% in the Single-Step Segmentation.

Fig. 6: ROC of Single-Step and Three-Step ground detection

Then, the ground truth was manually constructed for the recorded frames: each pixel of every image was labeled to be free floor, obstacle or unknown. The unknown category is needed because of the low resolution and noisy images of the Time-of-Flight camera, which making it impossible to label every part of the image correctly. This category also applies to every pixel where the camera has no valid measurement. Finally, the computed segmentation is compared pixelwise to the ground truth. Pixels that are categorized as *unknown* in the ground truth are not counted at all and have no contribution to resulting metrics.

Both the initial as the final segmentation were run on the data set using different threshold values. Figure 5 shows the resulting accuracy calculated over all 3700 frames for both algorithms. The accuracy is calculated as:

$$\text{Accuracy} = \frac{\# \text{True Positives} + \# \text{True Negatives}}{\# \text{Valid Pixels}}. \quad (3)$$

The best accuracy increased from 98.8% (using $t_1=100$ mm and $t_2=300$ mm) in Figure 5(a) to 99.4% (using $t_1=60$ mm and $t_2=180$ mm) in Figure 5(b). So apart from a significant increase in accuracy, an important decrease in optimal threshold values was obtained. This means that the thresholds can be chosen smaller so that tinier obstacles can be detected.

The Receiver Operating Characteristic (ROC) curve in Figure 6 plots the True Positive Rate against the False Positive Rate and the improvement of the final segmentation compared to the initial one. In particular, the True Positive Rate (recall) has increased.

Increased Robustness. Although the Single-Step Segmentation already showed a good overall accuracy, this accuracy dropped significantly in some specific frames where the camera was tilted with respect to the calibrated camera angle causing a high False Negative Rate. The robustness of a segmentation algorithm can best be observed in the worst-case frames. These are the frames with a relatively high False Negative Rate.

To validate the improved robustness, the frames with a False Negative Rate $> 10\%$ in the Single-Step Segmentation were selected for a comparison. In Figure 5(c) and (d) the accuracy of the selected worst-case frames can be compared. For example in the case where $t_1=60$ mm and $t_2=180$ mm there were 122 selected frames. The accuracy went from 60.4% in the Single-Step to 93.4% in the Three-Step Segmentation. An improved worst-case accuracy means an improved dependability in all situations.

4.2 Timing

Figure 7 and Table 1 show the required calculation time per frame in milliseconds. Matlab takes about double the time for the first frames because of initialization and memory allocation. Therefore, the calculation times for the first 10 frames are excluded.

Fig. 7: Calculation time for each frame

Table 1: Statistics of calculation time per 176×132 depth frame for 3690 frames. Calculations were executed in standard Matlab code on a 2.8 GHz Intel i7 quad core processor.

	[ms]	median	mean	std. dev.
initial segmentation		0.223	0.230	0.0272
ground plane update		0.800	0.801	0.0697
final segmentation		0.209	0.213	0.0148
total time		1.235	1.244	0.0919

All total frame times are well under 2 ms, averaging around 1.25 ms. About 65% of the time is spent in the update step calculating the best fit plane. This time could be further reduced by using an optimized solver or by using a sampled subset of prior free floor points in the plane fitting.

4.3 Indoor Results

The Three-Step Segmentation was also applied to the indoor data set recorded with a Kinect V2 ToF mounted on a wheelchair. No ground truth is available for this data, but visual examination shows a clear improvement. These improvements are situated mostly in the frames where the condition of planar motion was not sufficiently fulfilled.

The segmentation errors of the Single-Step Segmentation and the obtained improvement can be clearly seen in Figure 8 when the wheelchair accelerates or changes direction. In the last scene, the wheelchair is tilted because it is driving over a border. The angle error is too big to be fully corrected by the updated algorithm, but the segmentation is still slightly improved.

Fig. 8: Measurement results in various scenes based on Kinect V2 ToF measurements on a wheelchair. The left column shows the images of the RGB-camera, the middle column shows the initial segmentation results and the right column shows the final segmentation results. The red arrow indicates the acceleration of the wheelchair.

5 Conclusions

We set out to improve the simple Single-Step Segmentation algorithm [11] to correctly work on a mobile platform where the camera has more degrees of freedom than just planar motion. The resulting algorithm fulfills this requirement while retaining its simplicity and computational efficiency.

The experiment with our instrumented vehicle shows the accuracy of our free surface segmentation method applied on different road surfaces. The accuracy has increased from 98.1% to 99.4% with an obstacle detection threshold of 60 mm above the ground plane. For worst-case frames the accuracy improvement was even more prominent: from 60.4% to 93.4%, showing the robustness of the proposed segmentation.

While points near the vehicle are correctly segmented, the accuracy of discrimination of distant points is limited by the assumption that the road surface is a plane. A smooth road surface must be planar in the local neighbourhood of each point but is not necessarily planar globally. An improved algorithm could quantify this smoothness to segment sloping surfaces correctly.

Our experiment was limited by the currently available range cameras which are motion sensitive, have a limited range especially outdoors, and have a low resolution. Camera makers are working on forward-looking range cameras for the automotive industry that work well with fast movement and that have a long range.

References

1. Balta, H., De Cubber, G., Doroftei, D., Baudoin, Y., Sahli, H.: Terrain traversability analysis for off-road robots using time-of-flight 3d sensing. In: 7th IARP International Workshop on Robotics for Risky Environment-Extreme Robotics, Saint-Petersburg, Russia (2013)
2. Hansard, M., Lee, S., Choi, O., Horaud, R.: Time-of-Flight Cameras, SpringerBriefs in Computer Science, vol. 95. Springer London, London (2013)
3. Venturelli, M., Borghi, G., Vezzani, R., Cucchiara, R.: From depth data to head pose estimation: A siamese approach. In: Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP, (VISIGRAPP 2017). pp. 194–201. INSTICC, ScitePress (2017)
4. Meers, S., Ward, K.: Head-pose tracking with a time-of-flight camera. In: Australasian Conference on Robotics & Automation (2008)
5. Bostelman, R., Hong, T., Madhavan, R.: Obstacle detection using a time-of-flight range camera for automated guided vehicle safety and navigation. *Integrated Computer-Aided Engineering* 12(3), 237–249 (2005)
6. Tao, Z., Bonnifait, P., Frmont, V., Ibaez-Guzman, J.: Mapping and localization using gps, lane markings and proprioceptive sensors. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 406–412 (Nov 2013)
7. Meier, E.B., Ade, F.: Object detection and tracking in range image sequences by separation of image features. In: IEEE International Conference on Intelligent Vehicles. pp. 176–181 (1998)

8. Ess, A., Leibe, B., Schindler, K., Van Gool, L.: A mobile vision system for robust multi-person tracking. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. pp. 1–8. IEEE (2008)
9. Kwon, J., Dragon, R., Van Gool, L.: Joint tracking and ground plane estimation. *IEEE Signal Processing Letters* 23(11), 1514–1517 (2016)
10. Mufti, F., Heinzmann, J., Mahony, R.: 4D Ground Plane Estimation Algorithm for Advanced Driver Assistance Systems. INTECH Open Access Publisher (2011)
11. Van Crombrugge, I., Mertens, L., Penne, R.: Fast free floor detection for range cameras. In: Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 4: VISAPP, (VISIGRAPP 2017). pp. 509–516 (2017)
12. Kircali, D., Tek, F.B.: Ground Plane Detection Using an RGB-D Sensor. In: INFORMATION SCIENCES AND SYSTEMS 2014. pp. 69–77 (2014)
13. Torr, P.H., Zisserman, A.: Mlesac: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding* 78(1), 138–156 (2000)
14. Schwarz, L.A., Mateus, D., Lallemand, J., Navab, N.: Tracking planes with Time of Flight cameras and J-linkage. In: 2011 IEEE Workshop on Applications of Computer Vision (WACV). pp. 664–671. IEEE (jan 2011)
15. Rodríguez, A., Yebes, J.J., Alcantarilla, P., Bergasa, L., Almazán, J., Cela, A.: Assisting the Visually Impaired: Obstacle Detection and Warning System by Acoustic Feedback. *Sensors* 12(12), 17476–17496 (dec 2012)
16. Jimnez, D., Pizarro, D., Mazo, M., Palazuelos, S.: Modeling and correction of multipath interference in time of flight cameras. *Image and Vision Computing* 32(1), 1–13 (2014)
17. Fuchs, S.: Multipath interference compensation in time-of-flight camera images. In: Pattern Recognition (ICPR), 2010 20th International Conference on. pp. 3583–3586. IEEE (2010)