2010 IEEE International Conference on Robotics and Automation
Anchorage Convention District
May 3-8, 2010, Anchorage, Alaska, USA

Using Time-of-Flight Cameras with Active Gaze Control for 3D Collision Avoidance

David Droeschel, Dirk Holz, Jörg Stückler, and Sven Behnke

Abstract— **We propose a 3D obstacle avoidance method for mobile robots. Besides the robot's 2D laser range finder, a Time-of-Flight camera is used to perceive obstacles that are not in the scan plane of the laser range finder. Existing approaches that employ Time-of-Flight cameras suffer from the limited field-of-view of the sensor. To overcome this issue, we mount the camera on the head of our anthropomorphic robot *Dynamaid*. This allows to change the gaze direction through the robot's pan-tilt neck and its torso yaw joint.**

The proposed obstacle detection method is robust against kinematic inaccuracies and noise in the range measurements. The gaze controller takes motion blur effects into account and controls the gaze depending on the robot's motion and the obstacles in its vicinity.

In experiments, we demonstrate that our approach enables the robot to avoid obstacles that the laser range finder can not perceive. We also compare our active gaze control strategy with a fixed gaze orientation.

I. INTRODUCTION

Obstacle avoidance is an elementary capability for autonomous mobile robots to safely navigate in dynamic environments. For this task, 2D laser range finders are the most popular sensors. However, such a sensor provides only a two-dimensional distance profile of the environment in its scan plane and hence objects below or above the scan plan can not be perceived. To overcome this issue, sensor modalities are required that gain dense three-dimensional measurements of the environment.

Time-of-Flight (ToF) cameras provide such information. They are compact, lightweight, solid-state sensors which measure depth to reflective surfaces at a high frame rate and are therefore ideally suited for mobile robots. They employ an array of LEDs that illuminate the environment with modulated near-infrared light. The reflected light is received by a CCD/CMOS chip for every pixel in parallel. Depth information is acquired by measuring the phase shift of the reflected light. The use of ToF cameras has been studied in various fields of robotics, also for obstacle avoidance [1], [2]. Main limitations of this sensor are its limited measurement range, measurement inaccuracies, and its restricted field-of-view (FoV).

To overcome the sensor's limited FoV, we propose a 3D obstacle avoidance method that incorporates active gaze control to focus attention to the most relevant regions. We mount the camera on the head of our anthropomorphic robot Dynamaid [3]. This enables the robot to change the gaze direction through its pan-tilt neck and its torso yaw joint.

All authors are with the Autonomous Intelligent Systems Group, University of Bonn, Germany. Email: droeschel@ais.uni-bonn.de

Fig. 1. The anthropomorphic service robot Dynamaid [3]. The robot has an anthropomorphic upper body with a yaw joint in its torso and a movable head on a pan-tilt neck. Besides the ToF camera on its head it is equipped with a Sick LMS 300 laser range finder on its base.

Besides the ToF camera, the robot is equipped with a Sick LMS 300 laser range finder (LRF) on its base. Fig. 2 shows a CAD drawing of the robot with the sensors and their complementary FoVs.

From the ToF camera's depth image, obstacle points are detected and composed to a *virtual scan*. Complementary to the measurements of the 2D LRF, this *virtual scan* is used to avoid obstacles that are not in the scan plane of the LRF. Such *virtual scans* can be easily incorporated into methods that have been designed for 2D LRFs.

This paper is organized as follows: Sec. II summarizes related work in the field of obstacle avoidance, especially with ToF cameras. In Sec. III and IV we describe our main contributions: a method for obstacle detection using ToF cameras and an approach to active gaze control. We evaluate our approach in experiments in Sec. V.

II. RELATED WORK

So far, 3D LRFs are mostly used for 3D perception ([4], [5], [6]) due to their high measurement range and precision. However, they suffer from low frame rates and wear of mechanical moving parts. For this reason, ToF cameras have attracted attention in the field of robotics since their invention nearly a decade ago.

One of the first robotic applications of ToF cameras was

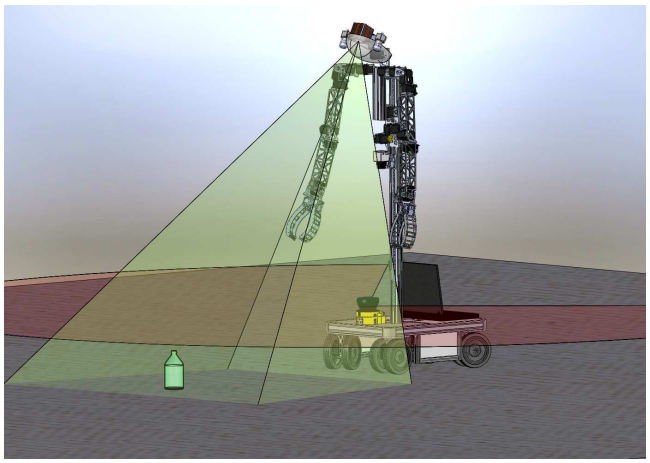


Fig. 2. CAD drawing of our robot and visualization of the field-of-views (FoV) of the sensors. The ToF camera measures dense depth in a narrow FoV (green cone), while the laser range finder measures in a scan plane (red). Small obstacles below the laser scanner's measurement plane can be perceived by the ToF camera. Its narrow FoV requires to actively control the gaze.

published in 2004. Weingarten et al. [1] used a CSEM SR-2 ToF camera prototype for basic obstacle avoidance and local path planning. They demonstrate that the use of ToF cameras improves obstacle avoidance. They mention that some objects are not perceived due to the limited FoV. Their camera was mounted statically on the robot.

Also, Yuan et al. [2] applied a ToF camera for obstacle avoidance. They used the camera to build a *virtual laser* [7] which is used in addition to the laser scan of a laser range finder for obstacle avoidance. Their approach also suffers from the limited FoV of the sensor. Besides mounting the ToF camera in a fixed orientation, both approaches use a simple height filter to segment measurements into floor and obstacle points. We detect points on obstacles at local height peaks.

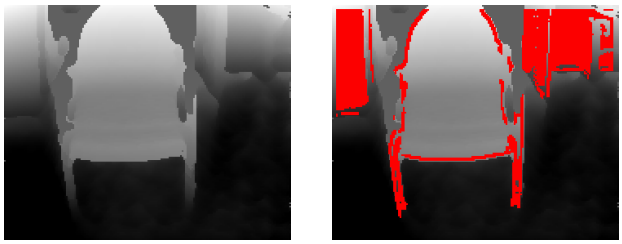
Seara et al. [8], [9] have proposed a gaze control scheme for their visually guided humanoid robot. The cameras are mounted on a movable head. Their approach to active gaze control arbitrates two concurrent objectives, i.e. obstacle avoidance and self-localization.

III. SENSOR DATA PROCESSING

Our approach to obstacle detection proceeds in three main steps: In a pre-processing stage, we filter mismeasurements out of the ToF data. Then, we detect points on obstacles in the filtered data, especially at the object boundaries. The last step builds a *virtual scan* from the obstacle points.

A. Filtering

Measurements of ToF cameras are subject to several error sources [10]. From the image, we filter out measurements with low amplitude, as these indicate either highly noisy measurements of poorly reflecting objects or measurements of objects beyond the ambiguity range of the camera.



(a) (b)

Fig. 3. (a) The generated *height image*. The grayscale value of every pixel corresponds to the z-coordinate of the respective point in the point cloud. (b) The resulting obstacle points (red).

A further error effect are so-called jump-edges at object boundaries. They can be detected by examining local pixel neighborhoods. We detect jump-edges when two points approximately lie along the line-of-sight of the camera [11]. Since this jump-edge filter is sensitive to noise, we apply a median filter to the distance values beforehand.

B. Detection of Obstacle Points

For obstacle avoidance, the 3D range image needs to be segmented into points on the drivable floor and obstacles. Yuan et. al. [2] and Weingarten et. al. [1] simply threshold the height of a point above the floor plane to separate obstacles from the floor. However, measurement noise and kinematic inaccuracies result in erroneous segmentations. Instead, we consider the local neighborhood of a point in the range image for segmentation.

Fig. 4(b) shows a typical example of a filtered point cloud taken in an indoor scene. We transform the filtered depth measurements to the robot reference frame which origin we define in the center of the base in floor height. The colors of the points correspond to the distance of a point from the sensor, brighter colors relate to shorter distances. From this point cloud we build a *height image* as shown in Fig. 3(a). A point $\mathbf{p}_{i,j}$ is classified as belonging to an obstacle, if

$$(W_{\max} - W_{\min}) > \epsilon_H, \tag{1}$$

where $W_{\max}$ and $W_{\min}$ are the maximum and minimum height values in a local window W, spanned by the Moore neighborhood around $\mathbf{p}_{i,j}$. The threshold ϵ_H thereby corresponds to the minimum tolerable height of an obstacle. It needs to be chosen appropriately since it cannot be smaller than the sensor's measurement accuracy. Due to evaluating a point's local neighborhood, floor points are inherently not considered as obstacles. The result of this filter is shown in Fig. 3(b).

C. Extraction of Virtual Scans

From the set of obstacle points a two-dimensional *virtual scan* is extracted. The number of range readings, the apex angle, and the resolution of the *virtual scan* are determined by the ToF camera's specifications. For the SR4000, the number of range readings is 176, which is the number of

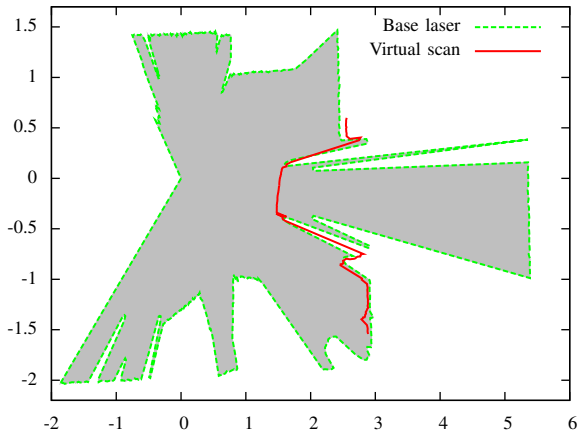


Fig. 5. Comparison of the resulting *virtual scan* (red line) of the scene with the scan from the laser range finder (dashed green line). In the base laser scan, only the chair legs are visible, whereas the *virtual scan* outlines the contour of the chair.

columns in the image array. The apex angle and the angular resolution are $43°$ and $0.23°$.

From every column of the ToF camera's distance image, the obstacle point with the shortest Euclidean distance to the robot is chosen. This distance constitutes the range in the scan. If no obstacle point is detected in a column, the scan point is marked invalid by setting it to the maximum range of the sensor.

Fig. 4(a) shows an example scene of an indoor environment. The point cloud which results from the ToF camera's depth image is shown in Fig. 4(b). The color of the points corresponds to the distance, brighter color relates to shorter distances and darker color to farther distances. The result of the filtering and the obstacle detection step is depicted in Fig. 4(c). Points with a low amplitude are removed from the cloud. Obstacle points are marked white and the obstacle points that contribute to the *virtual scan* are marked red. The remaining points are marked green.

The resulting *virtual scan* of the scene is compared with the scan from the laser range finder in Fig. 5. The base laser scan is illustrated by the dashed green line. The red line illustrates the *virtual scan*. The chair shows only a few points in the scan from the laser range finder, since only the legs of the chair are in the scan plane, whereas the *virtual scan* outlines the contour of the chair.

Similar to the base laser scan, the virtual scan is accumulated in an occupancy grid that is used by the local planner. The *forgetting* rate of measurements in the virtual scan depends on the orientation of the head. Measurements within the field-of-view of the ToF camera are forgotten faster than measurements outside the field-of-view. For this purpose, we calculate the view-frustrum for the current head orientation by *frustrum culling*, a technique that stems from 3D computer graphics [12].

IV. GAZE CONTROL

Compared to 2D laser range finders, the field-of-view of ToF cameras is rather limited ($43°$ vs. $>180°$). Practically, obstacles in the robot's immediate vicinity can only be perceived when lying directly along the line-of-sight. That is, all obstacles not falling into the robot's gaze direction form a potential source of collision. This poses the question on how to adapt the robot's gaze direction to keep all relevant regions in sight or at least to check, in regular intervals, whether or not the respective region can be traversed by the robot.

Keeping relevant objects in the sensor's limited field-of-view is the primary function of the proposed gaze controller. If no obstacle is present in the robot's immediate vicinity, it should orient the camera along its movement direction for being able to react to sudden dynamic changes, like for instance people passing by. However, if an obstacle is detected the robot should keep track of that obstacle in order to avoid collisions while still observing potential risks in its movement direction. That is, we need to adapt the gaze direction regularly by successively moving it from one relevant region to the next.

Analogous to changes in the gaze direction of the human eye, we refer to these kind of motions as *saccades*. We define the gaze direction as a vector $\mathbf{g} = (\mathbf{g}^x\ \mathbf{g}^y\ \mathbf{g}^z)^T$ representing a point in space that lies in the center of the sensor's field of view.

A. Gaze Directions

We distinguish between two kinds of gaze directions from which the controller can choose from – namely the *driving gaze* direction $\mathbf{g}_d$, and *obstacle gaze* directions $\mathbf{g}_o$ pointing towards closest obstacles.

1) Driving Gaze Direction $\mathbf{g}_d$: In order to keep track of obstacles in the robot's driving direction and for being able to perceive suddenly appearing obstacles, the gaze vector $\mathbf{g_d}$ corresponding to the driving direction solely depends on the current translational velocities $(\mathbf{v}^x\ \mathbf{v}^y)^T$ and the rotational velocity ω:

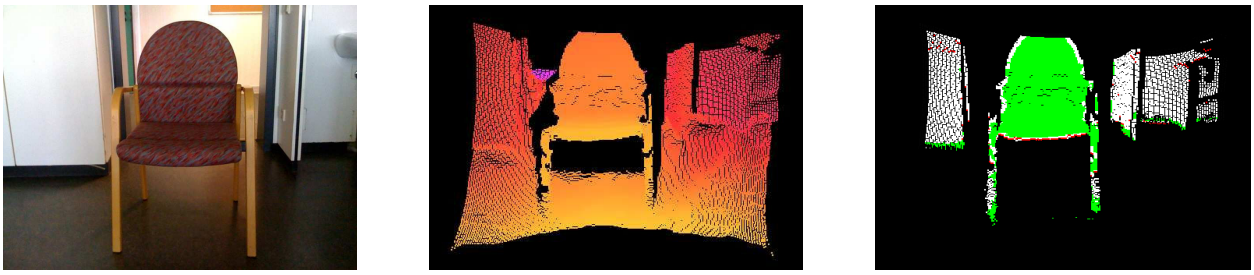
$$\begin{pmatrix} \mathbf{g_d}^x \\ \mathbf{g_d}^y \\ \mathbf{g_d}^z \end{pmatrix} = \alpha \begin{pmatrix} \cos\beta\omega & -\sin\beta\omega & 0 \\ \sin\beta\omega & \cos\beta\omega & 0 \\ 0 & 0 & 1 \end{pmatrix} \left(\frac{d_{\text{min}}}{\|\mathbf{v}\|} + \gamma \right) \begin{pmatrix} \mathbf{v}^x \\ \mathbf{v}^y \\ 0 \end{pmatrix}$$

where d_{min} is the minimum distance in front of the robot, that can be perceived. The constants α and β as well as the offset γ can be adjusted according to a specific robot platform. The offset $\gamma \geq 1$ can be adapted to prefer the perception of obstacles being farther away from the robot, e.g., when driving fast.

2) Obstacle Gaze Direction $\mathbf{g}_o$: For keeping track of closest obstacles not lying in the robot's driving direction, a gaze direction candidate $\mathbf{g_o} = (\mathbf{g_o}^x\ \mathbf{g_o}^y\ \mathbf{g_o}^z)^T$ is generated that points towards the closest obstacle in the occupancy grid.

B. Saccade Selection

Depending on the distance $\mathbf{d}$ to the closest obstacle and the time $\mathbf{t}$ that the obstacle was last detected in the field-of-view, the gaze controller chooses either $\mathbf{g_d}$ or $\mathbf{g_o}$ as gaze vector.



(a) (b) (c)

Fig. 4. (a) An example scene of an indoor environment. (b) The point cloud which results from the ToF camera's depth image. The color of the points corresponds to the distance, brighter color relates to shorter distances and darker color to farther distances. (c) The result of the filtering and the obstacle detection step. Points with a low amplitude are removed from the point cloud. Obstacle points are marked white and the obstacle points that contribute to the *virtual scan* are marked red. The remaining points are marked green.

$$\mathbf{g} = \begin{cases} \mathbf{g_o}, & d \leq \epsilon_\mathbf{D} \; and \; t \geq \epsilon_\mathbf{T} \\ \mathbf{g_d}, & \text{otherwise.} \end{cases} \tag{2}$$

where $\epsilon_\mathbf{D}$ is the distance threshold that defines an obstacle as *close* and $\epsilon_\mathbf{T}$ is a time threshold. The time threshold $\epsilon_\mathbf{T}$ prevents the robot from keeping the gaze fixated at a close obstacle. In the following experiments $\epsilon_\mathbf{D} = 1m$ is used.

C. Motion Blur and Dropping Frames

Since the acquired sensory information is heavily affected by motion blur when rotating the camera, we keep track of the head orientation and neglect depth images when considerable changes are detected. Furthermore, we keep the gaze direction constant for a time T after each saccade, where T corresponds to the sensor's data acquisition rate. That is, we a) wait until the camera is no longer rotated and then b) wait until at least one depth image has been captured in this static setup.

V. EXPERIMENTS AND RESULTS

To evaluate the proposed approached, we set up a minimal benchmark environment of size $7m \times 4m$. In each experiment, the robot is commanded to move from one side of the area to the other. In order to test the obstacle detection mechanism and the active gaze control, a set of objects has been used to setup a test scenario. We evaluated our approach in three individual experiments. What is examined in these experiments is the applicability of the obstacle detection mechanism and its integration into, respectively, local path planning and reactive collision avoidance.

The setup for this experiment series is the following: Two obstacles are placed in front of the robot with a distance of $180cm$ between each other. The first object is a white cubic box with a side length of $10cm$. The second object is a beverage can with a diameter of $5cm$ and a height of $10cm$. Both objects do not interesect the two-dimensional measurement plane of the laser range finder in a height of $27cm$ and require for 3D information in order to get detected.

Fig. 6 shows the result of the first experiment where the robot was solely using the geometric information acquired with the 2D laser range finder (dashed blue lines). As expected, the laser range scan accurately represents the environmental structures intersecting its scan plane whereas not a single measurement has been taken on the surface of one of the test objects (black circles). As a consequence, the robot collides with the objects as can be seen in the plotted trajectory (red line). Instead of swerving around the obstacles, the robot takes the shortest path leading it directly through the obstacles.

In the second experiment, the information from both sensors is used. That is, the local path planner is fed with both the 2D laser range scan and the *virtual scan* extracted from the 3D camera data. That is, we use the obstacle detection mechanism from Section III but not the active gaze control. Again, the robot first takes the direct path to the goal. That is, it moves a straight line until the first object gets into the field-of-view of the SwissRanger camera. The first object is detected and the robot successfully avoids first collision. However, by making the detour, the robot is laterally oriented to the second object while following its path. This causes that the second object does not get into the sensor's field of view and the robot collides. The trajectory of the robot as well as data from the different sensor modalities is shown in Fig. 7(a).

In the final experiment, both the obstacle detection using the ToF camera and the active gaze control are used. It is expected that by making multiple saccades during the robot's movement, both the closest objects in its vicinity as well as the obstacles appearing in its driving direction are detected and avoided. As is shown in Fig. 7(b), the robot adapts its trajectory and the followed path respectively. That is, both obstacles are successfully detected. The green vectors reflect the behavior of the gaze controller, switching between the robot's movement direction and adjacent regions of the environment. As soon as an obstacle is detected the robot tries to keep track of it, by periodically re-checking the respective region. If a previously detected obstacle has not been in the field-of-view for a certain time, the robot makes a saccade switching between the driving direction $\mathbf{g}_d$ and the respective object gaze vector $\mathbf{g}_o$. The resulting trajectory shows that the combination of the obstacle detection mechanism together with the active gaze controller allow for adequatly detecting

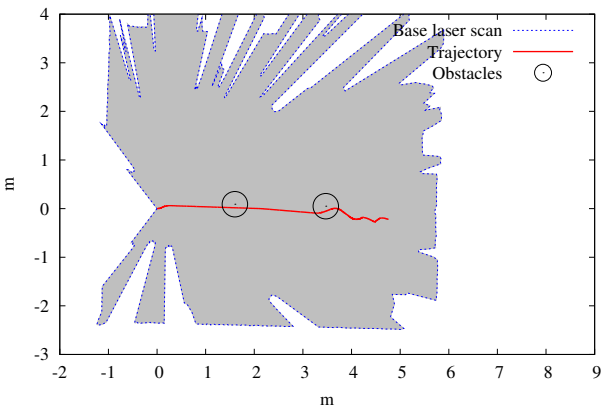


Fig. 6. The robot's trajectories with the laser range finder solely. The dashed blue line depicts the scan of the laser range finder mounted on the robot's base. The two obstacles are depicted by the black circles. Since the objects are below the scan plane the robot collides with them, driving a straight trajectory to the goal.

and reacting to obstacles.

VI. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

In this work we present an approach for 3D obstacle avoidance using a Time-of-Flight camera. With this sensor, our method can perceive obstacles that could not be measured in the scan plane of the laser range finder. The ToF camera is mounted on the head of our robot *Dynamaid* which allows to actively change the orientation of the sensor. It extends previous work, where the camera was mounted in a fixed orientation.

The proposed obstacle detection method is robust against kinematic inaccuracies and noise in the range measurements. The gaze controller takes motion blur effects into account and controls the gaze depending on the robot's motion and the obstacles in its vicinity.

In experiments we demonstrate that the robot is able to avoid obstacles that are not perceived by the laser range finder. The experiments have been carried out with a fixed gaze orientation and our active gaze control strategy which orients the sensor depending on the robot's driving direction and the distance to obstacles. A fixed gaze orientation lets the robot collide with an obstacle that it approaches laterally. In contrast, the active gaze control lets the robot avoid the obstacle.

Our approach is mainly limited by the sensor's inaccuracies in depth measurements. Especially on a poorly reflecting floor, small objects cannot be distinguished from the floor. Another limitation is the motion blur effect and the resulting data acquisition delay between saccades that limit the robot's performance, i.e. the maximal rotational and translational driving velocities.

B. Future Work

The experiments show that this simple gaze controller generates gaze vectors that are probably redundant. Since each saccade consumes time due to the duration the actuators need to orient the sensor and the delay that is necessary to minimize motion blur, a more sophisticated gaze controller could be formulated. In future work, a gaze controller that predicts an optimal gaze orientation for a given situation has to be investigated, maximizing the information gain for the possible gaze orientations.

REFERENCES

[1] J.W. Weingarten, G. Gruener, and R. Siegwart. A state-of-the-art 3d sensor for robot navigation. In *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 3, pages 2155–2160 vol.3, Sept.-2 Oct. 2004.

[2] Fang Yuan, A. Swadzba, R. Philippsen, O. Engin, M. Hanheide, and S. Wachsmuth. Laser-based navigation enhanced with 3d time-of-flight data. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pages 2844–2850, May 2009.

[3] J. Stückler and S. Behnke. Integrating Indoor Mobility, Object Manipulation, and Intuitive Interaction for Domestic Service Tasks. In *Proc. of the IEEE Int. Conf. on Humanoid Robots (Humanoids)*, 2009.

[4] S. Thrun, W. Burgard, and D. Fox. A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 321–328, San Francisco, California, USA, 2000.

[5] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, and P. Mahoney. Stanley: The Robot that Won the DARPA Grand Challenge. *Journal of Field Robotics*, 23(9):661–692, 2006.

[6] Hartmut Surmann, Kai Lingemann, Andreas Nüchter, and Joachim Hertzberg. A 3D Laser Range Finder for Autonomous Mobile Robots. In *Proceedings of the 32nd International Symposium on Robotics (ISR)*, pages 153–158, April 2001.

[7] O. Wulf, K O. Arras, H. I. Christensen, and B. Wagner. 2D Mapping of Cluttered Indoor Environments by Means of 3D Perception. In *Proceedings of the IEEE/RAS International Conference on Robotics and Automation (ICRA)*, pages 4204– 4209, New Orleans, Louisiana, USA, 2004.

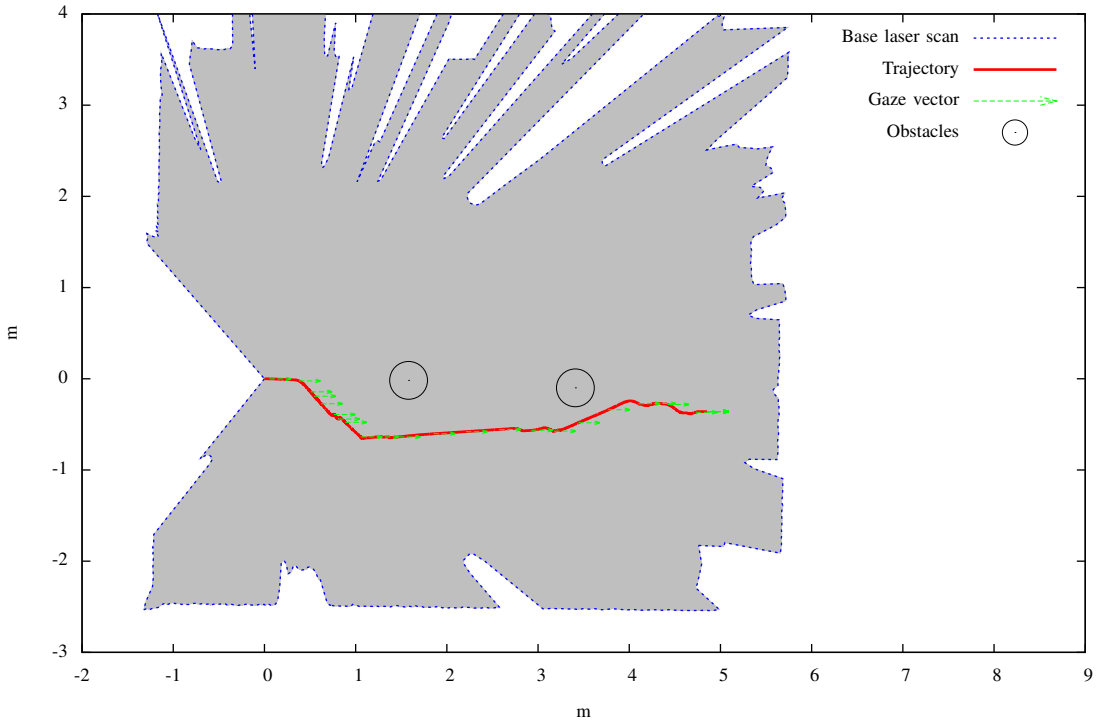
[8] J.F. Seara and G. Schmidt. Intelligent gaze control for vision-guided humanoid walking: methodological aspects. *Robotics and Autonomous Systems*, 48(4):231 – 248, 2004. Humanoids 2003.

[9] J.F. Seara, K.H. Strobl, and G. Schmidt. Path-dependent gaze control for obstacle avoidance in vision guided humanoid walking. In *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, volume 1, pages 887–892 vol.1, Sept. 2003.

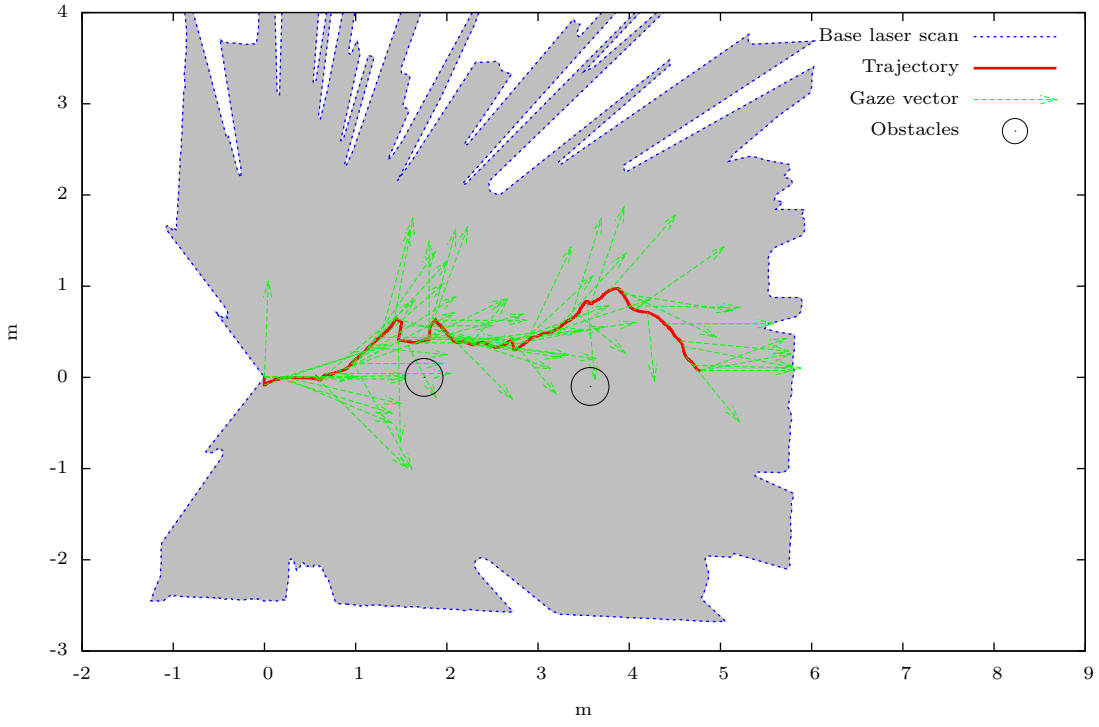
[10] Robert Lange. *3D time-of-flight distance measurement with custom solid-state image sensors in CMOS/CCD-technology*. PhD thesis, University Siegen, 2000.

[11] S. May, D. Droeschel, D. Holz, S. Fuchs, and A. Nüchter. Robust 3D-Mapping with Time-of-Flight Cameras. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, St. Louis, Missouri, USA, 2009. To appear.

[12] Eric Lengyel. *Game Programming Gems*, chapter A Fast Cylinder-Frustrum Intersection Test. Charles River Media, August 2000.



(a)



(b)

Fig. 7. The robot's trajectories, using the information from the *virtual scan* is fused with the scan from the laser range finder. In (a) the camera orientation (green arrows) is fixed. The robot avoids the first obstacle, but collides with the second obstacle. (b) shows the results of our experiment with active gaze control. The resulting trajectory shows that the robot navigates around the obstacles without a collision.