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

Vision-based Localization Using an Edge Map Extracted from 3D Laser Range Data

Paulo Borges[1], Robert Zlot[1], Michael Bosse[1], Stephen Nuske[2], and Ashley Tews[1]

Abstract— **Reliable real-time localization is a key component of autonomous industrial vehicle systems. We consider the problem of using on-board vision to determine a vehicle's pose in a known, but non-static, environment. While feasible technologies exist for vehicle localization, many are not suited for industrial settings where the vehicle must operate dependably both indoors and outdoors and in a range of lighting conditions. We extend the capabilities of an existing vision-based localization system, in a continued effort to improve the robustness, reliability and utility of an automated industrial vehicle system. The vehicle pose is estimated by comparing an edge-filtered version of a video stream to an available 3D edge map of the site. We enhance the previous system by additionally filtering the camera input for straight lines using a Hough transform, observing that the 3D environment map contains only linear features. In addition, we present an automated approach for generating 3D edge maps from laser point clouds, removing the need for manual map surveying and also reducing the time for map generation down from days to minutes. We present extensive localization results in multiple lighting conditions comparing the system with and without the proposed enhancements.**

I. Introduction

Research on the automation of industrial vehicles has received significant attention in recent years. Heavy industry can benefit from automated mobile equipment by increasing productivity and improving personnel safety. Accurate localization technologies are not only critical for autonomous navigation, but can also be instrumental in the optimization of operations with manned vehicles.

To further this goal, we investigate a vision-based localization system applied to a large forklift-type vehicle (Fig. 1). This type of vehicle is primarily used in the aluminum industry to transport molten metal around smelters and is therefore called a hot metal carrier, or HMC. HMCs operate for long periods of time both indoors and outdoors in changing lighting conditions; therefore, any visual localization solution must be able to handle varying illumination conditions.

Ultimately, the objective of our research is to develop a fully dependable autonomous industrial vehicle. To achieve dependability, the accuracy and robustness of the localization system are essential. Standard solutions such as GPS are not applicable due to indoor navigation requirements and multi-path effects near buildings. Other sensor modalities, including laser [1] and WiFi [2], have also been explored for our localization requirements. Our strategy to achieve robustness is to employ multiple localization systems that can be fused compared in order to automatically detect failure or degradation of one of the systems. By choosing different sensing modalities, the likelihood of a simultaneous failure of all systems is low, despite there being the potential for individual failures. For example, the aluminum production process requires strong DC currents, which result in extremely high magnetic fields in parts of the operational space. These fields are quite capable of rendering equipment with small moving parts, such as a 2D laser scanner, inoperable. The aim of the vision-based localization system discussed in in this paper is to act as an independent system to increase localization reliability.

Fig. 1: The autonomous hot metal carrier.

An approach for vision-based localization is to autonomously build visual maps consisting of image point features, such as SIFT features [3], [4], [5]. One challenge, however, when building such maps, is that the features are not invariant to non-planar scenes [6]. Another significant concern is the fact that the image features can change dramatically according to lighting conditions, especially in the case of outdoor operations [3].

To overcome these issues an alternative strategy is to compare an edge-filtered version of the input from the camera with a predefined 3D edge map of the environment. Several authors have proposed such an approach in the computer vision literature. Drummond and Cipolla [7] and Reitmayr and Drummond [8] apply a real-time edge localization technique indoors and outdoors. In those works, however, the 3D edge-based techniques calculate only a single pose estimate

[1] – Autonomous Systems Laboratory; CSIRO ICT Centre
Brisbane, Australia
Firstname.Lastname@csiro.au
[2] – The Robotics Institute; Carnegie Mellon University
Pittsburgh, PA, USA
nuske@cmu.edu

each iteration, which makes the system more susceptible to error. Multi-modal techniques such as the recent particle filter method developed by Klein and Murray [9] offer improved robustness by maintaining many pose estimates per frame.

The method proposed in this paper further extends our previously published particle filter approach [10] which is based on the Klein and Murray algorithm. In our industrial site, the architectural configuration of the buildings is permanent, and therefore provides a good set of salient features from which to localize. We choose line features at the edges and intersections of walls, roofs, and doorways as they provide a robust representation and it is possible to observe them from both monocular camera images and 3D laser point cloud data.

The site has been professionally surveyed and a 3D edge map of the environment has been created by measuring the end points of linear features on major building structures. In the first incarnation of the localization system, this surveyed edge map was required for the observation model. In addition to the time and financial cost of professional surveying, it is also sometimes not possible to get significant access to parts of an industrial site both to avoid interruptions in operation and placing humans into hazardous areas. In the proposed version, we eliminate the requirement of contracting surveyors by extracting the edge set autonomously from 3D laser point cloud data. Our other main contribution is the use of the probabilistic Hough transform as a post-processing step to filter out non-linear edges from the camera input to the localization algorithm. This modification improves the performance of the algorithm by reducing the number of unsuitable detections used as observation candidates. The proposed modifications are evaluated under several different illumination conditions, during operations at different times of the day and under different weather conditions.

The remainder of this paper is organized as follows. In Section II we describe in detail the visual localization system. We then introduce technique to improve the robustness of the system with the use of the Hough Transform to filter the edge set in Section III. In Section IV we describe a method to automatically generate the edge map using 3D data collected from two laser range sensors. We provide results from experiments in our industrial site in Section V, followed by our conclusions in Section VI.

II. System Description

The proposed vision-based technique uses cameras mounted on a vehicle tracking linear features such as building edges, doors, and rooflines in a large outdoor industrial building environment. For this task, a sparse 3D edge map of the site is utilized, consisting of around 20 large industrial buildings. This map can be generated via professional surveying, or acquired automatically with laser range sensors, as described later in Section IV. Examples of the surveyed map and of the 3D laser generated map are given in Fig. 4. Once the map is created, a vehicle moving through the environment can be localized by matching edges in the map with edges extracted from the camera images. The comparison between the image and the map is calculated for each pose hypothesis in a particle filter and provides a likelihood measure for that particle.

Full details on the particle filter implementation can be found in a previous publication [10]. This reference also provides additional information about the system, such as robustness to occlusions and intelligent exposure control to deal with challenging outdoor lighting conditions.

III. Application of the Hough Transform

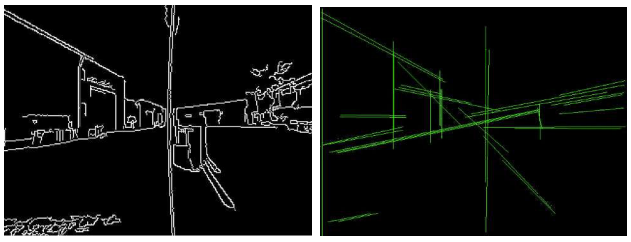
In this section we propose the use of the Hough transform to improve the performance of the algorithm. The Hough transform [11] is a well-known and effective method for finding lines fitting a set of 2D points. Each line is represented by two parameters, ρ and θ, which represent the length and angle from the origin of a normal to the line. These parameters are discretized and measurement data (*e.g.*, image pixels) are accumulated into bins in both distance and angle space.

The use of the Hough transform has been proposed in numerous applications in image and video processing [12], including robot localisation. For example, Kim *et al.* [13], employ the Hough transform to detect curbs in an autonomous driving application, whereas Chang *et al.* [14] utilize the transform to detect walls in an occupancy grid.

In our system, the 3D edge models consists solely of straight lines such as roofs, walls, and doors, making it suitable for the use of the Hough transform. For this reason we employ the transform as a post-processing step after the input image has been edge-detected and binarized. This enhancement reduces the effect of noise in similarity matching between the image and the 3D edge model, where noise is considered any non-straight element such as trees or clouds, for example.

Let $H(\mathbf{x})$ represent to the Hough transform line filtering operation on an edge binary image $\mathbf{x}$. Correspondingly, let $D(\mathbf{x})$ represent the morphology dilation operation [12]. The final edge image $\mathbf{y}$ used by the localization algorithm is given by $\mathbf{y} = D(H(\mathbf{x})) \otimes \mathbf{x}$, where $\otimes$ represents the element-wise AND logical operation. This operation reduces the amount of lines that are erroneously picked by the Hough transform that do not correspond to the ones in the 3D edge model. The dilation avoids cases where very small angular misalignments between the edge pixels in $\mathbf{x}$ and $H(\mathbf{x})$ cause a significant part of the relevant edge pixels in $\mathbf{x}$ to be discarded.

In our implementation, we use the probabilistic Hough transform [15]. The probabilistic version of the transform considers that if just a small subset of the edge points in the image are selected at random, the performance is only slightly impaired, with a considerably shorter execution time. Fig. 2 illustrates the application of the Hough transform to one of the edge images extracted from our experiments. The Hough transform improves the overall performance of the system, at the expense of increased computational complexity.



(a) Original edge-filtered image. (b) Edge-filtered image after straight line detection via Hough transform.

Fig. 2: Illustration of the application of the Hough transform to the edge images. The long vertical edge in the middle of the figure corresponds to sun glare in a bright day.

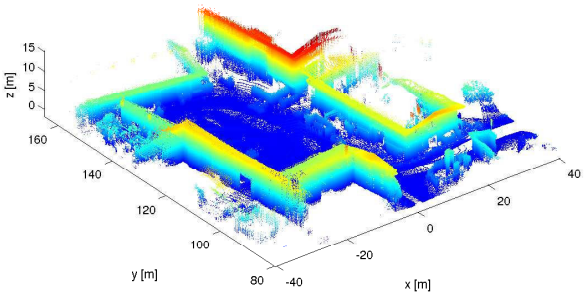


Fig. 3: A 3D point cloud of the test area. Colors indicate height, with blue at the ground transitioning to red at the maximum height.

IV. Extracting the Edge Map from 3D Laser Data

The initial implementation of the edge-based localization system [10] relied on the availability of a surveyed 3D edge map of the environment. A team of professional surveyors was hired for this task, and the resulting edge map contained linear features at the edges of buildings, along rooflines, and around garage doors (Fig. 4a). Even with a fairly small site, the survey required two full days of work. Despite its accuracy, the survey was limited to a particular set of requested features, and ignored others that may be visible in a camera image. Any changes to the environment at a later time would have to be remeasured to update the surveyed map. Given these disadvantages, we propose that an automated laser-based edge extraction algorithm would be a more flexible and cost-effective approach.

Laser data was acquired by mounting two SICK LMS 291 laser scanners on one side of the HMC with their scan planes vertical and oriented at a right angle relative to one another. This configuration was chosen to reduce the effects of occlusion. However, it also requires the vehicle to traverse its path in both directions to ensure full coverage since the 3D point cloud data is collected only on one side. Accurate vehicle positioning is required to accurately register the 3D points, and is achieved using a laser-based SLAM algorithm. Data from the vehicle's four horizontally-mounted lasers are first used to compute an 2D approximation to the trajectory [16], which is then taken as a prior for the 3D registration of the vertical laser scan data. A registered 3D point cloud of the test environment is shown in Fig. 3.

A. Line Extraction

Two types of 3D lines are extracted from the 3D point cloud of laser range returns. The first are lines at the intersection of extracted planes, the second type are edge lines linking depth discontinuities. The plane intersections are best for finding lines between the ground and walls of a building or between adjacent sides of a building. The plane intersection lines are more accurate and less affected by measurement noise since they are supported by many more points than the edge-fitted lines. 3D lines can be indirectly extracted from the intersections between adjacent planar segments that are found in the point cloud. However, plane intersections are not able to find the lines on the tops of the buildings or around open doorways. Therefore, lines are also fitted to laser points at depth discontinuities in order to extract additional linear edge features.

1) Plane Intersection Edges: To determine the plane intersection edges, planar surfaces are first segmented from the point cloud. The local neighborhood of each point is used to classify its local shape. A kd-tree is used to find the 10 nearest neighbors of each point within a fixed radius of 2 m from which a covariance matrix is computed. The eigenvalues of this covariance matrix are analyzed to determine the local shape. If the smallest eigenvalue is significantly smaller then the second smallest, the local shape is considered planar and the eigenvector corresponding to the smallest eigenvalue indicates the surface normal.

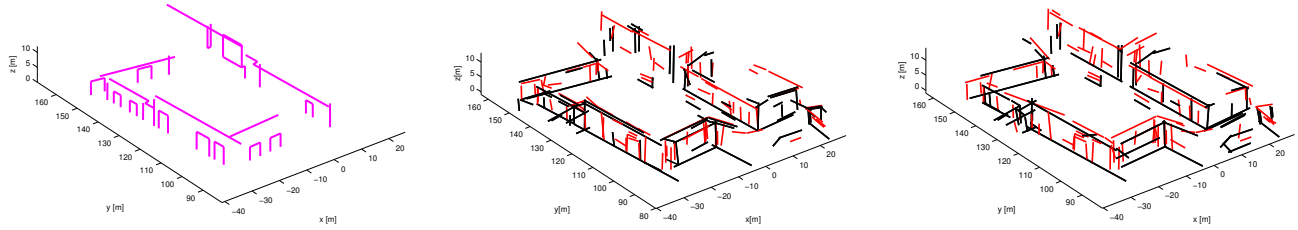
A bottom-up segmentation procedure iteratively merges planar points into clusters provided that their normals are similar. The plane fitting a cluster of points can be determined from its moments up to second order; *i.e.*, the mean and covariance of all the points in the cluster. The smallest eigenvector is the plane normal and the centroid is used to determine the plane's offset. To build the clusters, pairs of adjacent points are processed in order of their decreasing shape similarity. If the points are in separate clusters that appear to be from the same plane, then the clusters are merged. The merging decision is based on the plane alignment error ϵ, which is computed as a weighted combination of the plane offset errors d_o (measured along the normals) and the difference in normal angle d_n:

$$d_o(\Pi_a, \Pi_b) = [(\mu_a - \mu_b)^T n_a]^2 + [(\mu_a - \mu_b)^T n_b]^2 \quad (1)$$

$$d_n(\Pi_a, \Pi_b) = 1 - (n_a^T n_b)^2 \quad (2)$$

$$\epsilon(\Pi_a, \Pi_b) = w_o d_o + w_n d_n \quad (3)$$

where μ_a, μ_b are the centroids and n_a, n_b are the normals for planes Π_a and Π_b. The weights w_o and w_n are chosen at appropriate scales such that the criterion for merging is that the alignment error is below a threshold value of 3. The new cluster's second order moments can be computed directly from the moments of the two previous clusters without having to reanalyze all the original points. The plane



(a) Edge map generated from survey data.

(b) Laser-extracted edge map: $w_n = 6/\pi$, $w_o = 2.5m^{-2}, c_{min} = 0.925$, $\lambda_{max} = 0.09m, l_{min} = 0.33m$.

(c) Laser-extracted edge map: $w_n = 4/\pi$, $w_o = 2m^{-2}, c_{min} = 0.9$, $\lambda_{max} = 0.09m^2, l_{min} = 0.33m$.

Fig. 4: The surveyed edge map and two edge maps automatically extracted from the 3D laser data. In the extracted edge maps, black lines indicate plane intersection edges and red lines indicate range discontinuity edges. The two extracted edge maps differ in the set of parameters utilized, and are not necessarily intended to represent the same set of edges present in the surveyed model.

segmentation procedure is complete when all the adjacent point pairs have been examined.

The planar clusters with a sufficient number of points are considered for line extraction. The pairs of adjacent planes are determined from the adjacent point pairs which belong to separate planes. If the angle between the surface normals of adjacent planes is large enough (about 10°) then the intersecting line is calculated. The endpoints of the line are computed from the minimum and maximum extents of the points from the planes when projected orthogonally onto the line. The edge is included in the edge map if the distances between the line and both plane boundaries are below a threshold d_p.

2) Depth Discontinuity Edges: The procedure for extracting lines from edges in the point cloud is very similar to the plane extraction method. First the local neighborhoods of each point is used to test if a lidar point lies on a potential edge. If there is a depth discontinuity in the lidar scan at the point or the local curvature is at a maximum the point is labeled an edge. Adjacent edge points are then clustered using the same procedure as the plane clustering but with a different set of metrics. The largest eigenvector of the cluster's covariance matrix determines the line direction, and the endpoints of the line are determined from the maximal and minimal projections of the points onto the line. Only clusters with a sufficient number of points and a cylindrical shape are retained as extracted edge lines. The cylindrical shape is represented by a parameter $c = \frac{\lambda_3 - \lambda_2}{\lambda_1 + \lambda_2 + \lambda_3}$ computed from the eigenvalues $\lambda_1 \leq \lambda_2 \leq \lambda_3$ of the cluster covariance matrix. The value of c must be above a threshold c_{min} for the line to be included in the edge set. A threshold λ_{max} on the maximum value for λ_2 is also used to eliminate non-cylindrical clusters. These thresholds are only applied beyond the point where the candidate edge has grown beyond a minimum length l_{min} to compensate for the larger effects of noise on nascent clusters.

3) Edge Post-Processing: After extracting the edges, horizontal edges on the ground that occur due to ground strikes and point cloud boundaries are filtered from the edge set. A manual culling step is then used to remove edges that appear due to laser returns from objects within the buildings which would not be visible in the camera images. Other edges that are on the far sides of buildings that would not be visible in the video stream can also be removed similarly. While the presence of these edges does not tend to degrade the algorithm's accuracy in our environment, their removal does improve computational efficiency. More generally, for larger and more complex environments, occlusion of such edges should be taken into account, and can be partially handled by considering the directions of the normals from the surfaces that generated the edges when matching with the video data.

4) Related Work: A similar classification of edges for extraction for 3D point cloud data has been described by Stamos and Allen [17] and Stamos and Leordeanu [18]. Their plane-fitting algorithm [17] differs in that they allow for a smoothly varying normal direction, thereby fitting some curved surfaces. As these types of edges do not generally translate to straight lines in the edge-processed image (and such curved surfaces do not appear in our test environment), our model did not need to consider such edges. By combining the moment statistics as the plane is expanded, additional robustness is provided when fitting to completely flat surfaces. Nonetheless, a region growing heuristic similar to that described by Stamos and Allen may be more appropriate in environments with more complex surface shapes.

Plane intersection has been previously used for line extraction from both ground-based [17] and aerial [19] platforms. Stamos and Leordeanu [18] further describe a method for extracting edges at the extreme points of planar surfaces rather than utilizing range discontinuities. Gao *et al.* [20] include only vertical lines in their application, which are obtained by comparing the eigenvector corresponding to the largest eigenvalue of a point cluster with a vector along the vertical.

B. Evaluation

Fig. 4 illustrates the surveyed edge map and two edge maps extracted from the laser point clouds. The difference between the two laser-generated maps is the set of thresholds used when growing the regions during the extraction process. To extract an edge set from the raw data takes on the order of minutes depending on the size and density of the point cloud. The examples in Fig. 4 require approximately ten minutes of processing.

Despite the existence of a ground truth edge map from the survey, it is difficult to definitively evaluate the accuracy of a laser-generated edge map for several reasons. Firstly, the survey map does not represent a complete list of all relevant edges from the environment. Rather, it contains a particular subset of linear features (namely building edges, rooflines, and garage door edges) that the surveyors were instructed to measure. Many other linear features in the environment could be picked up by the image processing algorithm which are equally valid. Therefore, some 'false positives' (with respect to the survey data) extracted from the 3D point cloud may in fact be valuable edges; while others, caused by noise or other algorithmic limitations, may be disruptive. Secondly, a few missed edges from the survey map will not necessarily result in poor performance either. With fewer available edges to match to, the localizer may lose some accuracy; but the effect would be minimal if there are a sufficient number of remaining edges. A limitation of using range sensors is that many texture discontinuities visible in the camera images do not correspond to range discontinuities, and therefore will not be picked up by the line extraction algorithm. Moreover, some salient edges are difficult to detect in our 3D data. For example, a garage door may have been closed during data collection, or in some cases the region around the edge contained a significant amount of clutter. Finally, a surveyed edge might be split into more than one edge in the laser edge map, which in many cases will not affect performance. Generally, the surveyed edges and the autonomously extracted edges simply represent a different set of linear features in the environment; but in both cases the visual localization system can operate successfully.

We have created a set of heuristics and metrics to be of general assistance in coarsely evaluating the accuracy of the extracted edge map and choosing appropriate parameters for the 3D line detection algorithm. First, by simply visually comparing the surveyed and extracted maps, a qualitative assessment can be conducted (Fig. 4). Second, we can get a general sense of accuracy by comparing those edges that correspond with edges in the surveyed model. These similarity metrics are based on three criteria: the shortest distance between corresponding line segments d_l; the difference in angle between them d_α; and the relative difference in segment length d_s (*i.e.*, the ratio of the absolute length difference to the length of the surveyed edge). Rather than combining these three quantities into a single score (*e.g.*, Jiang *et al.* [21] apply arbitrary weights to similar metrics), we instead prefer to analyze the distribution of these errors separately. We compute a correspondence between each surveyed edge and an edge in the extracted set by finding the closest match according to each of the metrics provided the error is below a threshold ($d_l \leq 1.5\,\mathrm{m}$, $d_\alpha \leq 7.5°$, $d_s \leq 2$). The thresholds provide a conservative limit within which we can consider matched edges to have likely come from the same structural feature in the environment. The resulting correspondences are not highly sensitive to the choice of thresholds: a reasonable range of values within the expected operating tolerance produces the same error distributions. Fig. 5 shows the similarities between the survey edge set and three different extracted sets (obtained from the same point cloud data using different parameters in the line extraction algorithm). The quality of each edge set can be analyzed by observing frequency of small errors relative to larger errors and the total count in each distribution. Here we observe that with a good choice of parameters, the location and angle of the corresponding extracted edges typically have low error (less than 50 cm and 3°), but we occasionally have large discrepancies in the lengths of the segments.

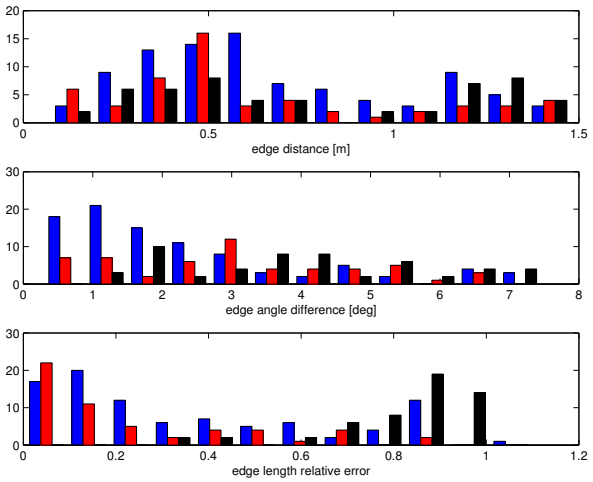


Fig. 5: Histograms of similarity scores between three extracted edge sets and the surveyed edge map. The errors are only computed for edges that have correspondences in the survey set. The distribution for each extracted edge set is indicated by a different color. The edge map indicated with blue bins corresponds to the map in Fig. 4b and the red color corresponds to Fig. 4c. For comparison, the results in black demonstrate an example of poorly chosen parameters resulting in an edge set with few good matches.

A more suitable test of the edge map quality is carried out by comparing a set of candidate models on the full localization system and evaluating the resulting pose error. We are able to easily run and assess the system on logged data played back in real-time. In the section that follows, we evaluate a set of extracted edge maps in this manner.

V. Experiments

Experiments evaluating the localization system are divided into two categories. The first set of experiments corresponds to the use of the surveyed edge map. The second set investigates the performance of the system when edge maps autonomously generated from 3D laser data are used. Implementation and hardware details are given next.

A. Experimental Setup

1) The Standard Mission: In order to have fair and meaningful comparisons among the performance of the algorithm in different illumination conditions, we create a "standard mission". We use this mission as a reference to record the video data and test the algorithm in different times of the day and weather conditions. This mission is a predetermined trajectory around our industrial site, performed autonomously via a highly accurate 2D laser-based localisation system [1].

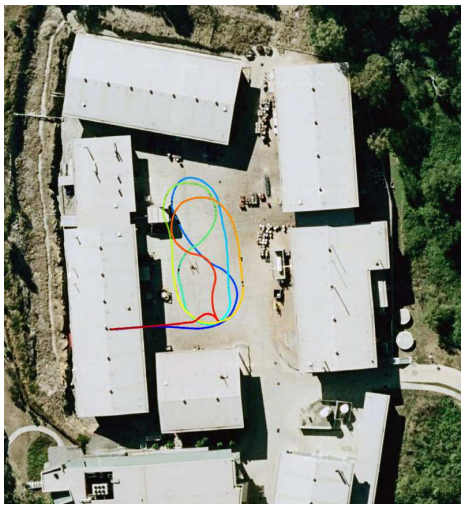


Fig. 6: The standard mission path. The trajectory is colored by time: blue at the start transitioning to red at the end.

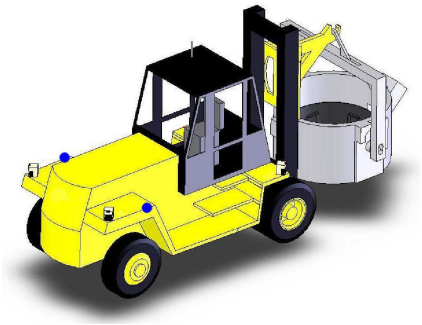


Fig. 7: Illustration of the camera setup mounted on the HMC. The blue circles indicate the camera positions. Each is pointed outwards perpendicular to the direction of travel.

A plot with the standard mission path around the site is shown in Fig. 6. A more detailed representation with scale is given by the red curve in Fig. 9a. The total distance travelled in the standard mission is 1620 m, which is performed in approximately 7 minutes, covering most areas and orientations within the test environment.

2) Camera Setup: The tests are performed with two firewire digital cameras (Basler scA780-54f), with resolution of 780×580 pixels. The cameras are mounted facing sideways relative to the direction of vehicle motion, at the locations illustrated by the blue dots in Fig. 7. Both cameras are fitted with Kowa LM4NCR (3.5mm F1.4 2/3") lenses. The camera setup is calibrated and verified by ensuring that the edges from the model projected onto the image plane from known vehicle poses are aligned with the image edges. This is illustrated in Figs. 8a and 8b, where both distorted and undistorted versions of the input image are shown, along with the overlapped edge model.

3) Particle Filter Parameters: The parameters of the particle filter are determined following the methodology described by Nuske *et al.* [10]. The approach used to calculate these parameters is to record video, odometry, and ground truth laser pose data from the vehicle when it performs the standard mission. The particle filter is then run off-line several times through the same sequence of recorded data to optimize these parameters. This optimization is achieved by varying the parameters at each iteration and comparing the average pose error estimate. Although a large number of particles ideally improves the performance of the filter, it also increases the computational complexity. Operating at 10 Hz, we set the number of particles to 1000 to achieve a satisfactory trade-off between error and speed.

B. Surveyed Edge Model Results

In the first set of experiments we employ the surveyed edge map (Fig. 4a) for comparison with the camera data, as described in Sec. II. Table I shows the average estimated positional error for different times of the day and weather conditions, measured relative to the 2D laser-based localizer, taken as ground truth. The corresponding results are under the "Surveyed" columns in the table.

Fig. 9a illustrates the standard mission path (in red) overlaid with the vision-based localizer path (in blue). An example of how the vision-based localizer positional error evolves in time is given by the blue curve in Fig. 9b. In this figure, the green curve represents the vehicle odometry error, which is based on wheel encoders. The average positional error of the vision-based localizer was 0.447 m, with a peak error of 1.21 m. Correspondingly, the average odometry error was 2.71 m with a peak error of 11.23 m. In addition, the odometry system is open loop and subject to increasing error due to drift. In contrast, the visual localizer does not suffer from this problem as long as it is able to detect mapped features in the environment.

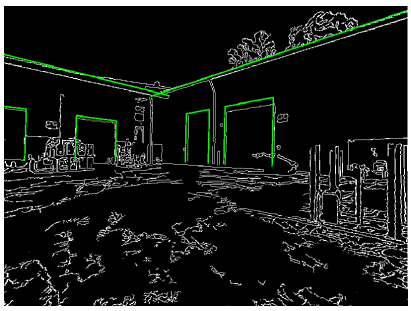
The average error results when the Hough transform filter is employed are shown in Table I, under the columns indicated by "Hough." In our implementation of the Hough transform we set the resolution parameters ρ to 1 and θ to $\pi/180$. The dilation operation D on the resulting image is performed with a 3×3 square structuring element and a single iteration. A plot illustrating the reduced error is given in Fig. 9c. The red curve represents the Hough transform case, whereas the blue line corresponds to case in which the transform is not used. Depending on the video instance, the transform provides a significant reduction in the estimated positional error, decreasing the average error to 0.389 m and the peak error to 0.890 m. Despite an increase of approximately 15% in the processing time when the transform is used, the system can still be run in real-time with the parameters described in Sec. V-A.

C. 3D Lasers Edge Model Results

The results presented in this section employ the 3D edge map autonomously generated with the vertical lasers rather than the surveyed one. As described in Sec. IV, several different parameters influence the quality of the generated edges. We test the system with different versions of the edge maps, which differ in number of edges and edge lengths, depending on the parameters used to generate them. Examples of edge maps generated with different parameters are

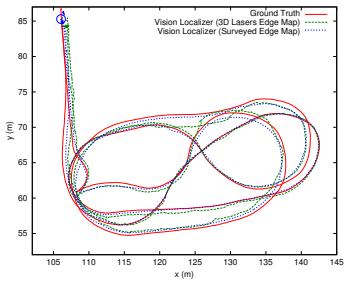
(a) Fish-eye image.

(b) Image after undistortion, overlaid with edge model (in green).

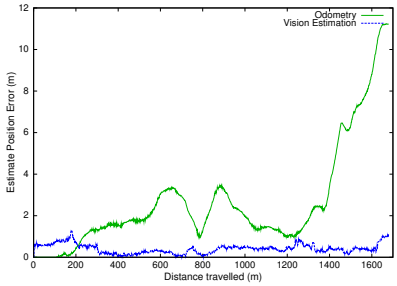


(c) Edge image after undistortion, overlaid with edge model (in green).

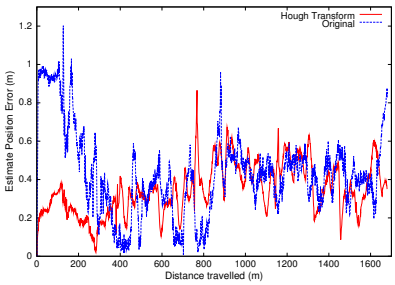
Fig. 8: Example of camera images and calibration.



(a) Trajectory estimates: comparison of vision-based and laser-based localization systems.



(b) Vision-based positional error (in blue) compared to the odometry error (in green).



(c) With and without Hough transform.

Fig. 9: Accuracy of the localization system under cloudy weather conditions.

TABLE I: Average positional error (in meters) for different parameters and weather conditions.

	Sunny				**Cloudy**			
	Surveyed		**Laser Extracted**		**Surveyed**		**Laser Extracted**	
	Non-Hough	Hough	Non-Hough	Hough	Non-Hough	Hough	Non-Hough	Hough
Early Morning	0.53	0.47	0.86	1.06	0.43	0.38	0.75	0.98
Mid-day	1.12	1.01	0.99	0.94	0.39	0.52	1.20	0.90
Mid-Afternoon	0.45	0.40	1.09	1.49	0.37	0.39	1.09	1.21
Late Afternoon	0.48	0.38	1.15	0.82	0.39	0.35	0.79	0.74
Dusk	0.53	0.45	0.88	1.08	-	-	-	-

given in Figs. 4b and 4c. The results given in Table I under the "Laser Extracted" fields represent the positional error for the edge set which yields the best observed performance (Fig. 4b), considering the laser localizer as ground-truth. Fig. 10 plots the positional errors for the highest accuracy edge map tested (in green) and for the lowest accuracy edge map tested (in red). The blue plot corresponds to the error obtained with the surveyed edge map. Using the highest accuracy edge map, the average error was 0.875 m with a peak error of 1.75 m. The resulting travel path is shown by the green curve in Fig. 9a. Correspondingly, for the lower accuracy edge map, the average error was 1.390 m and the peak error was 3.41 m. We note that the better performing edge map corresponds to the better performing map from Fig. 5; however, further experiments would be necessary to provide sufficient evidence of whether or not the similarity metrics predict localization accuracy.

The results when the Hough transform filter is employed are also shown in Table I under the "Laser Extracted - Hough" columns. In this case, the Hough transform does not improve the system performance significantly when compared to the gain observed when the surveyed edge map is used. We hypothesize that this result is partially due to the fact that the edge map is composed of smaller lines which are not perfectly contiguous, in contrast to the surveyed map.

An accompanying video demonstrates the localizer performance over a segment of the standard mission for both the surveyed and laser-extracted edge maps.

VI. Conclusions

We have presented a vision-based localization method for a ground vehicle operating in an industrial environment. The localizer is an extension of the work presented by Nuske *et al.* [10]. The system compares the edge images from onboard cameras to a pre-generated 3D edge map of the environment in order to perform pose estimation. The main advantages over the previous system is the use of the Hough transform for the detection of straight lines

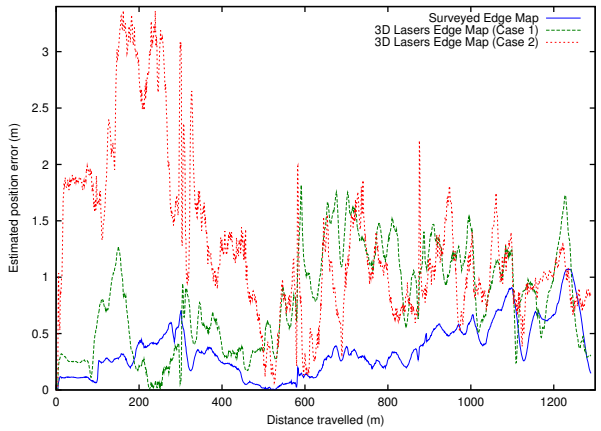


Fig. 10: Positional error comparisons between the surveyed edge map and edge maps generated with the 3D lasers without the Hough transform under cloudy weather conditions. The green curve corresponds to the edge map shown in Fig. 4b, while the red curve corresponds to Fig. 4c.

and the use of an autonomously generated 3D edge map. The use of a straight line filter is justified by the fact that linear structures are common in industrial environments. Experiments illustrate improved results over the previous system. The autonomously generated edge map proposed in this paper replaces the need for professional surveying, and demonstrates that usable edge maps can be generated with 3D laser range data. Even though the average error is slightly increased with the non-surveyed edge maps, their use is justified by the fact that professional surveying can be expensive, time consuming, and in some cases, infeasible. Using the laser-based system, we have been able to construct larger maps of our site than were previously available from the survey team.

We provide experimental results for different times of the day and weather conditions, illustrating the advantages of the proposed method. The experiments indicate that the system achieves the highest accuracy in overcast weather. In bright sunny conditions, effects from sun glare and shadows negatively affect the algorithm's accuracy, as often observed in outdoor computer vision. We have additionally tested the system during a severe dust storm which covered the east coast of Australia in September, 2009 (see Fig. 1). Due to the resulting uniformity in the illumination, the localizer performed with an even higher accuracy than when under cloudy conditions. Unfortunately, we were unable to compare the results for those tests as at the time there were shipping containers obstructing the standard mission trajectory.

In the near future, we will be integrating the vision-based localizer into our HMC localization system to provide an additional level of redundancy and reliability. The next stage of development will focus on field trials of the system at a larger industrial site. Preliminary results have demonstrated the effectiveness of our laser-based edge detection algorithm in the smelter environment. Vision-based localization trials using the full system will commence shortly.

ACKNOWLEDGMENTS

The authors thank Polly Alexander, Paul Flick, and Dave Haddon of the CSIRO Autonomous Systems Laboratory for their technical support and assistance in running the experiments.

REFERENCES

[1] J. Roberts, A. Tews, C. Pradalier, and K. Usher, "Autonomous hot metal carrier - navigation and manipulation with a 20 tonne industrial vehicle," in *IEEE International Conference on Robotics and Automation*, April 2007.

[2] F. Duvallet and A. Tews, "WiFi position estimation in industrial environments using Gaussian processes," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2008)*, 2008.

[3] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[4] N. Karlsson, E. D. Bernardo, J. Ostrowski, L. Goncalves, and P. Pirjanian, "The V-SLAM algorithm for robust localization and mapping," in *IEEE International Conference on Robotics and Automation*, April 2005.

[5] C. Weiss, H. Tamimi, A. Masselli, and A. Zell, "A hybrid approach for vision-based outdoor robot localization using global and local image features," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, October 2007.

[6] A. Vedaldi and S. Soatto, "Features for recognition: Viewpoint invariance for non-planar scenes," in *IEEE International Conference on Computer Vision*, 2008.

[7] T. Drummond and R. Cipolla, "Real time visual tracking of complex structures," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2002.

[8] G. Reitmayr and T. Drummond, "Going out: robust model-based tracking for outdoor augmented reality," in *International Symposium on Mixed and Augmented Reality*, 2006.

[9] G. Klein and D. Murray, "Full-3D edge tracking with a particle filter," in *British Machine Vision Conference*, 2006.

[10] S. Nuske, J. Roberts, and G. Wyeth, "Robust outdoor visual localization using a three-dimensional-edge map," *Journal of Field Robotics*, vol. 26, no. 9, pp. 728–756, September 2009.

[11] P. V. C. Hough, "Methods and means for recognizing complex patterns." in *U.S. Patent 3,069,654*.

[12] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. Prentice-Hall, 2002.

[13] S.-H. Kim, C.-W. Roh, S.-C. Kang, and M.-Y. Park, "Outdoor navigation of a mobile robot using differential GPS and curb detection," in *IEEE International Conference on Robotics and Automation*, April 2007.

[14] H. Chang, C. Lee, Y.-H. Lu, and Y. Hu, "Simultaneous localization and mapping with environmental structure prediction," in *IEEE International Conference on Robotics and Automation*, May 2006.

[15] N. Kiryati, Y. Eldar, and A. M. Bruckstein, "A probabilistic Hough transform," *Pattern Recognition*, vol. 24, no. 4, pp. 303–316, 1991.

[16] M. Bosse and R. Zlot, "Map matching and data association for large-scale 2D laser scan-based SLAM," *International Journal of Robotics Research*, vol. 27, no. 6, pp. 667–692, June 2008.

[17] I. Stamos and P. K. Allen, "Geometry and texture recovery of scenes of large scale," *Computer Vision and Image Understanding*, vol. 88, pp. 94–118, November 2002.

[18] I. Stamos and M. Leordeanu, "Automated feature-based range registration of urban scenes of large scale," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2003.

[19] J.-B. Lee, K.-Y. Yu, Y.-I. Kim, and A. F. Habib, "Segmentation and extraction of linear features for detecting discrepancies between LIDAR data strips," in *IEEE Geoscience and Remote Sensing Symposium*, 2005.

[20] C. Gao, M. Sands, and J. R. Spletzer, "Towards autonomous wheelchair systems in urban environments," in *International Conference on Field and Service Robotics*, July 2009.

[21] H. Jiang, C.-W. Ngo, and H.-K. Tan, "Gestalt-based feature similarity measure in trademark database," *Pattern Recognition*, vol. 39, pp. 988–1001, 2006.