Detection and Localization of Curbs and Stairways Using Stereo Vision

Xiaoye Lu and Roberto Manduchi
Department of Computer Engineering, University of California, Santa Cruz
{*xylu,manduchi*}*@soe.ucsc.edu*

***Abstract*— We present algorithms to detect and precisely localize curbs and stairways for autonomous navigation. These algorithms combine brightness information (in the form of edgels) with 3-D data from a commercial stereo system. The overall system (including stereo computation) runs at about 4 Hz on a 1 GHz laptop. We show experimental results and discuss advantages and shortcomings of our approach.**

I. Introduction

Robots that move autonomously in urban or indoor environment must be able to recognize certain types of features, be it to avoid them (obstacles), to move towards them (targets), or simply to build a symbolic representation of the world. This paper is concerned with the robust visual identification and precise 3–D localization of a ubiquitous class of features that we will collectively call "curbs". A curb in the context of this paper is not only the step separating a road surface from the sidewalk, but also any generic step (such as in a stairway), as well as the junction of a wall with the floor.

Detection and localization of curb lines are important for two reasons. First, a robot may not be able to negotiate a curb or, if it can, may need to align itself precisely with the curb edge. Second, tracking and following a curb line represents a simple and effective percept–referenced behavior for urban navigation.

Together with curb detection, we present a new algorithm for the localization and characterization of stairways. Again, stairways represent important landmarks. Some robots are able to climb a flight of stairs [18], and in this case it would be desirable to determine the slope of the staircase, the size of the steps, and the orientation and endpoints of the step edges.

Intuitively, curbs should not be difficult to detect if an accurate 3–D snapshot of the environment can be acquired, by means, for example, of a 2–axis laser rangefinder (ladar)[1]. But even the latest 3–D sensors are bulky, expensive and slow[2]. A stereo camera is an obvious alternative to a ladar. Stereo systems are inherently inexpensive, have small dimensions, and may provide 3–D data at full frame rate. A number of stereo devices are available commercially; for our work, we have used a popular product, the Videre STH-MDS/-C camera with baseline of 9 mm (www.videredesign.com).

Unfortunately, the quality of stereo data is often inferior to that of ladar measurements. We concentrate here on correlation–based stereo, since this is the technique used by the Videre system. An analysis of errors associated with searching for disparities using a correlation window was presented in [17]. It is well known that stereo does not work well with poorly textured areas. Note that a textured image patch may become textureless due to saturation, if camera exposure is not properly controlled. In addition, due to occlusions and to effects related to the finite size of the correlation window, range errors can be expected in correspondence of depth discontinuities. Imperfect calibration (due, for example, to slight mechanical misalignments produced by the robot's motion) is another source of errors, which is particularly harmful in the context of curb detection, as we elaborate in Section II-G.

Our experience has shown that, due to such reasons, using stereo range data alone would not produce acceptable detection results in general. Our strategy is thus to use information from both image and geometry, which, in a stereo system, are perfectly registered together (indeed, this is an added advantage of using stereo). In fact, even a single image conveys a good amount of information about the presence of a curb, since a curb line normally looks like a brightness edge. Several other edges, however, usually co–exist in an image, and thus brightness alone would not be sufficient for reliable detection. Our main effort in this work was to combine information from range and image edges in a robust way, in order to minimize the amount of false positive while ensuring good detection and precise 3–D localization.

II. Curb Detection

The main components of our curb detection system are shown in Fig. 1. Edge points are first extracted from one of the images in the stereo pair. Stereo range information is then employed for a number of different purposes: to determine the ground plane, which is then used to eliminate points that are unlikely to be part of the curb, as well as inconsistent curb edges; to compute a suitable curvature index, which is used to weight edge points in a Hough transform framework; and to

* This work was supported by DARPA–MARS2020 Program, through subcontract 1080049-132209 from Carnegie Mellon University.

[1] A single–axis ladar or even just a laser striper coupled with a camera [12] can be used to detect a curb reliably by analyzing a vertical slice of the scene. However, these devices cannot provide a full description of the curb in terms of orientation and visible endpoints, unless data is integrated as the robot moves around.

[2] An active 3–D sensors with limited dimensions and high frame rate (but low resolution) has been recently commercialized by Canesta (www.canesta.com). At this time, we don't have enough evidence to establish whether this sensor is suitable for the purpose of curb detection.

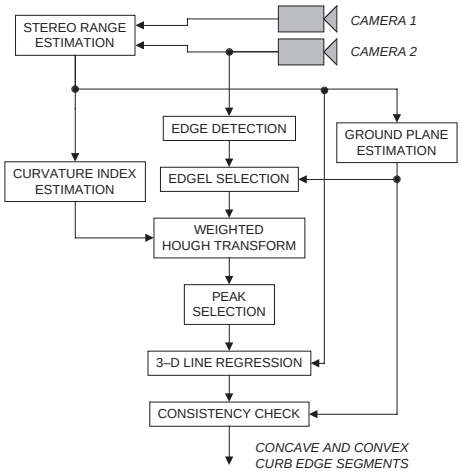

Fig. 1. A block scheme of our curb edge detector.

regress 3–D lines from the points that are back–projected from candidate lines in the image. The output of this algorithm is a set of 3–D curb edge segments, classified as concave or convex.

Stairway detection and modeling requires further steps (see Fig. 12). A plane is fitted using a robust procedure to each set of concave and convex step edges detected by the previous algorithm. The plane with the smallest residual error is then chosen, and the corresponding step lines are projected onto it. Outlier lines that are not parallel to most other lines are then rejected. This algorithm thus yields both the equation of a plane tangent to the stairway and a number of concave or convex step edge segments.

This paper is organized as follows. After reviewing previous work in Section II-A, we present in Section II our algorithm for curb detection and localization, together with some experimental results and a discussion of the shortcomings of our current approach. In Section III we introduce our technique for modeling and localizing staircases. Section IV has the conclusions.

A. Previous Work

There has been extensive work over the past fifteen years in the field of vision–based lane and road detection for autonomous vehicles (see e.g. [2], [3], [4]). However, this work mostly considered problems such as lane marker and obstacle detection, and did not specifically address the detection of curbs.

Curb detection over short distances for safe driving has been demonstrated at CMU with a laser striper [12]. The problem with a fixed laser striper is that the viewing geometry is very limited, while our task requires the ability to detect features over a rather wide field of view. Closer to our work is the paper by Se and Brady [10]. They detect candidate curbs by finding clusters of lines in an image using the Hough Transform. Then, in order to classify a curb line as a step–up or step–down, they compute the ground plane parameters of the two regions separated by the curb line. This allows one to precisely estimate the height of the curb. In contrast with the early–commitment approach of [10], which uses only image–based information to detect curb lines (and can give erroneous detection in cluttered environments with many distractors), we combine edge-based and range–based information to reliably localize curbs in the scene. The work of Turchetto and Manduchi [16] used the weighted Hough Transform for curb edge localization, with weights values proportional to the scalar product of the brightness gradient and of the depth gradient in the image. With respect to [16], the present paper uses a more sophisticated weight function based on a descriptor of the surface curvature. In addition, robust statistical techniques are used for the 3–D regression of line segments in space.

For what concerns stairway detection and modeling, all of the existing work we are aware of (e.g., [11], [18]) considers monocular analysis, and basically searches for pencils of lines in the image that correspond to the perspective projection of parallel lines in 3–D space. Since we have range data available, we can work directly in 3–D, which greatly simplifies model fitting and outlier rejection.

B. Ground Plane Estimation

The first module in our system is a routine that estimates the visible ground plane. Ideally, in the case of planar surfaces, a curb could be identified by the points that are at a certain distance from the ground plane. However, this strategy does not work in practice, for two reasons. First, road surfaces are often not planar (although this could be accounted for by more complex surface models). Second, this approach would rely solely on elevation data which, in the case of stereo, are often inaccurate and noisy.

Knowledge of the ground plane is used to rule out incorrect candidate curb lines. More precisely, search is constrained to points that are within a certain distance H_{GP} from the ground plane (e.g., H_{GP} = 0.5 meter). In addition, a candidate line is rejected if it forms an angle of less then θ_{GP} with the ground plane normal (e.g., $\theta_{GP} = 45°$).

To estimate the ground plane we use the Least Median of Squares (LMedS) robust procedure on the range data from stereo [9]. If the attitude of the camera is known at least approximately (e.g., from the IMU of the robot), then guided sampling [13], [14] can be used to increase the likelihood of sampling inliers points. The idea of guided sampling is simple: rather than sampling the search space uniformly, the density of samples is increased in those regions where the plane is more likely to be. Unfortunately, in our system, attitude information was not accessible continuously, although the inclination of the stereo rack (which was kept vertical, see Section II-F) when the robot was not moving could be measured. This was used to estimate a "default" plane relative to the camera position. In addition, the maximum expected pitch and roll angles were known, which defined the set of possible ground planes with respect to the camera. Before running LMedS, we normally discard all range points beyond a certain distance (e.g., $D_{\max}$ = 5 meter) as well as all

points that do not belong to the volume spanned by the set of possible ground planes. This simple procedure effectively rejects a good amount of points that are likely to be outliers. Note that points beyond $D_{\max}$ are then re–inserted after ground plane estimation.

The ground plane is estimated at each frame. Once a plane is found by LMedS, it is compared with the "default" ground plane as well as with the ground plane estimated in the previous frame. The plane among such three candidates that best fits the data is then declared the ground plane for the current frame.

C. Curvature Estimation

Surface curvature is obviously a good indicator for a potential curb edge (i.e., a flat surface cannot be an edge). Thus, it may seem that differential singularities such as ridges [5] of the range data could be used directly to detect candidate points. However, we found that a ridge operator applied to stereo data often gives unsatisfactory results. Spurious ridges are likely to appear, even when the range data has been smoothed. In addition, the "correct" ridges are often not rectilinear, due to noise in the range data. In contrast, differential singularities computed on the image (brightness edges) are more likely to faithfully correlate with actual curb lines. Our strategy is thus to rely on brightness edgels for the localization of candidate curb edge points, and on a curvature field defined over all range data to weight such edgels.

The question now is how to assign a meaningful "curvature index" (*CI*) value to each range point. One may expect that, in the vicinity of a curb edge, one of the two *principal curvature* values k_1 and k_2 will be large (in magnitude), while the other one will be small [15]. The proposed curvature index, defined in terms of k_1 and k_2, is defined as follows:

$$CI = \frac{(|k_1| - |k_2|)^2}{(|k_1| + |k_2|)^2 + \epsilon} \tag{1}$$

where ϵ is a small positive constant. One notices that $0 \leq CI < 1$; CI is equal to 0 when the two curvatures are identical, and becomes close to 1 when one principal curvature is small (in magnitude) while the other one is large (see Fig. 2).

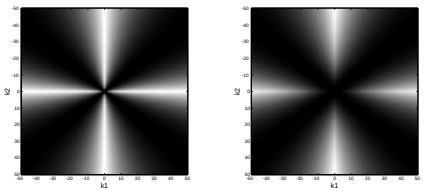

Fig. 2. The proposed curvature index *CI* as a function of the two curvature indices k_1 and k_2 (see (1)) for $\epsilon = 1$ (left) and $\epsilon = 100$ (right).

Although the estimation of principal curvatures requires computing square roots (a computationally expensive operation), this is not the case for our chosen curvature index. This is seen by re–writing (1) in terms of the *mean curvature*, $H = (k_1 + k_2)/2$, and of the *Gaussian curvature*, $K = k_1 k_2$:

$$CI = \begin{cases} \frac{H^2 - K}{H^2 - \epsilon} & , K \geq 0 \\ \frac{H^2}{H^2 - K - \epsilon} & , K < 0 \end{cases} \tag{2}$$

The expressions for H and for K are reported in [15], and involve estimating the first, second and partial derivatives of the measured surface with respect to x and y. Gaussian smoothing is performed on the range data before computing such derivatives; the constant ϵ in (1) was set to 1 in our experiments. Examples of curvature index fields for real images are shown in Figs. 4 and 5.

D. Line Selection

Candidate curb lines are selected starting from image edgels (edge segments) corresponding to 3–D points that are closer than H_{GP} to the ground plane. A weighted Hough transform [6] is used to detect such lines, with the weights derived from the curvature index CI at each point. Using the Hough transform ensures that a curb line can be detected even when only a limited number of edgels are found on it.

A large value of CI signifies that an edge point is likely to belong to a curb line. However, one should consider also the case when range is not available (for whatever reason) for some of the pixels. Accordingly, we proposed the following weight function at each edge pixel:

$$w = \alpha CI + (1 - \alpha) \tag{3}$$

where α is a positive constant between 0 and 1. The term $(1 - \alpha)$ ensures that some weight is assigned to edge points even when range is not available. When stereo data is reliable, α can be set to a value very close to 1; in our experiments, we set α to 0.95.

The peaks of the weighted Hough transform identify candidate curb lines. A potential problem with using weights is the risk of detecting multiple lines intersecting at one point with large weight. This is because a single point "votes" according to its weight for a whole pencil of lines. To reduce the risk of these concentric line patterns being detected, we use a non–maxima suppression strategy: a point in the Hough transform (parameterized by slope and intercept) is considered a peak only if it has the highest value within a certain window around itself. In this way, concentric lines that have very similar slope or intercept are avoided.

E. 3–D Line Regression

The procedure in the previous section allows one to estimate a number of candidate curb lines defined over the image plane. The next two steps are: (1) Regression of 3–D lines from the back–projection of the image points in the detected image lines; (2) Determination of the endpoints (in 3–D) of such curb segment. We'll begin by describing the first task.

Let's assume that a large enough number N_i of points in a given image line l_i have range values (otherwise, the candidate line l_i is discarded). Such points can thus be back–projected onto known 3–D points in the scene. Let us call

the back–projected 3–D points $\{p_j\}$. Simple geometry shows that all 3–D points in $\{p_j\}$ must lie on the same plane, Π_i, containing the focal point and the image line l_i. A Cartesian basis for this plane can be easily found either based on the intrinsic camera parameters, or directly from the 3–D point coordinates using SVD [15]. Thus, 3–D line regression of these 3–D points boils down to 2–D regression[3] in the plane Π_i. Care must be taken in this operation, since a large number (possibly, more than one half) of structured outliers (i.e., points belonging to other planar surfaces) can be expected. This precludes the use of traditional robust methods such as LMedS or RANSAC, and calls for the use of the Least K–th of Squares (LKS) algorithm [1] instead. In addition, one can rely on the weights computed by (3) for guided sampling. This is because one may reasonably expect that points in Π_i which do not belong to a curb line should have small curvature index CI and therefore small weight.

Our regression procedure, which is a simple variation of LMedS, is detailed in the following:

1) Let $\{\bar{p}_j\} \subset \{p_j\}$ be the subset of 3–D points in $\{p_j\}$ that are the back–projections of edge points in the image.
2) Sample (without replacement) two points from $\{\bar{p}_j\}$. A generic point $\bar{p}_j$ is chosen with probability $Z \cdot w_j$ (w_j being the weight of point $\bar{p}_j$ as defined in (3) and Z being a normalization constant). Determine the line through the two sampled points.
3) Compute the distances (residuals) d_j of each point in $\{p_j\}$ (including non–edge points) to this line. Compute the weighted residuals $\hat{d}_j = d_j / w_j$, and determine the K–th smallest such residual.
4) Iterate steps 2–3 for a fixed number of times. Return the line with the smallest associated K–th smallest weighted residual, $\hat{d}^{(K)}_{\min}$.

Guided sampling reduces the probability of sampling a point outside the curb line. The choice of weighted distances $\hat{d}_j$ increases the likelihood that points with small curvature index are assigned to the outlier subset. Note that this procedure assumes that there are at least K inliers, and therefore K must be sufficiently small. Algorithms to determine a suitable value for K have been proposed in [8], [7]. In the interest of computational speed, we used a fixed value of $K = 0.2 \cdot N_i$. An example of robust line fitting in Π_i is shown in Fig. 3 (left).

Once the 3–D curb line (lying on Π_i) has been regressed, the next task is the determination of the endpoints of the 3–D curb edge segment. This is a typical problem in robust statistics, and requires the estimation of a suitable "scale" in order to separate inliers from outliers. A simple strategy would be to threshold all of the weighted residuals $\{\hat{d}_j\}$ in the previous algorithm by a value proportional to $\hat{d}^{(K)}_{\min}$. A more elaborate approach is the Modified Selective Statistical Estimator (MSSE) proposed in [1], whereby an unbiased scale estimate, σ_n, is computed from the smallest n–th residuals. The index n is increased (starting from $n = 2$) until the $(n+1)$–th smallest residual is larger than $T\sigma_n$ (where T is a positive constant). Then, all points with $\hat{d}_j > T\sigma_n$ are considered outliers. We have experimented with both methods, obtaining comparable results.

[3]Note the early–commitment character of this operation: an error in the image line detection cannot be recovered by subsequent 3–D reasoning.

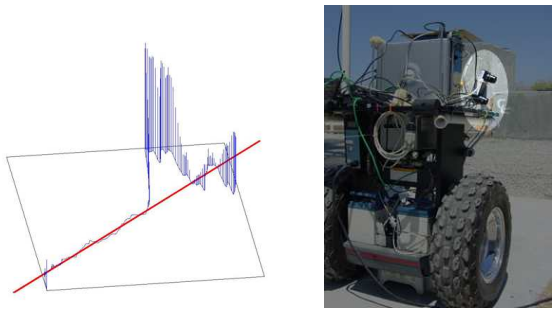

Fig. 3. Left: The points $\{p_j\}$ in Π_i and the regressed line. The height of each point marker is proportional to the weighted residual $\hat{d}_j$ of the point with respect to the line. Right: The Videre stereo camera used for our experiments seen on Stanford's Segbot.

In addition to this procedure, we use simple heuristics to enforce spatial coherence on the points belonging to the same line. First, small gaps between segments of points that survived the previous steps are filled; then, small segments are removed. Finally, 3–D segments that are inconsistent with our model are rejected. These are: (1) segments that form an angle of less than θ_{GP} with the ground plane normal, and (2) segment pairs whose projections on the image intersect each other. The first constraint was already discussed in section II-B. The second constraint derives from the fact that it is topological impossible for the projection of two curb lines to intersect anywhere else than at an endpoint. Finally, once the line segments have been determined, each segment is labeled as "concave" or "convex" based on the sign of the average mean curvature of its points.

F. Experiments

Our algorithms run under Linux on a 1GHz laptop. We use Videre's Small Vision System (SVS) software for stereo matching and Intel's Open Computer Vision library for a number of subroutines. Curb detection is performed at a rate of approximately 4 Hz (including disparity computation over 320 by 240 pixels). To maximize the expected angle between curb lines on the image and the epipolar lines (see Section II-G), the stereo rig was oriented vertically, as in Fig. 3 (right).

Successful examples of curb detection are shown in Figs. 4–8. In some cases, the 3–D data is shown from behind to improve readibility. Figs. 4–5 also display the detected image edges and the curvature index *CI* field. The estimated ground plane is visible in Figs. 5–7. Two examples with inaccurate detection are shown in Figs. 9 and 10. In the case of Fig. 9, the position of one endpoint is grossly incorrect. This may happen at times due to failure of the scale estimation module for outlier detection (Section II-E). Another problem is shown in Fig. 10, where a line on the floor was mistakenly

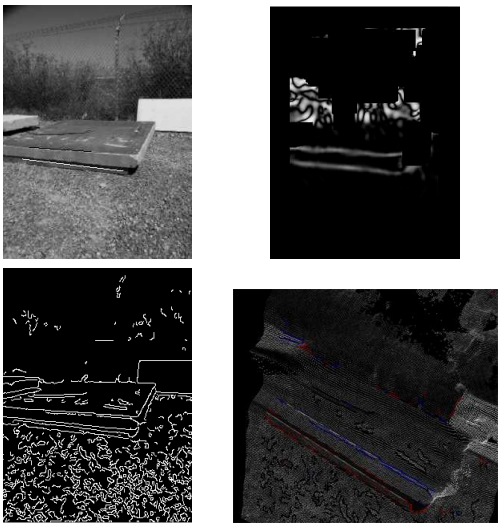

Fig. 4. Top left: The left image in a stereo pair, with the detected curb lines (white: concave; black: convex). Top right: The curvature index *CI* field, defined only for points that have range measurements. Pixels for which the value of *CI* could not be computed are marked in black. Bottom left: The brightness edges obtained by Canny's edge detector. Bottom right: The brightness edges mapped onto the reconstructed 3–D surface. Colored edges have a large value of the associated curvature index CI. The color of an edgel depends on the sign of its associated mean curvature (blue: negative; red: positive).

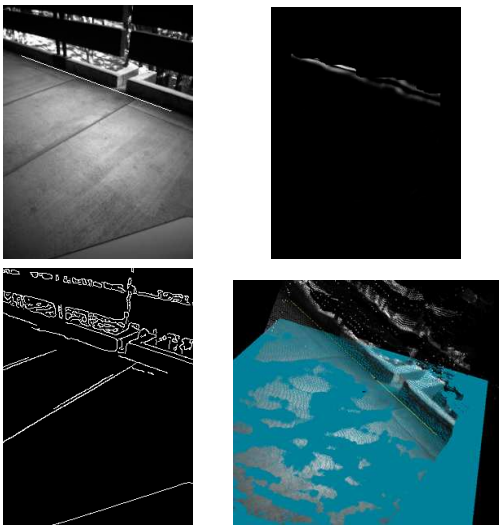

Fig. 5. Top left: The left image in a stereo pair, with the detected concave curb line. Top right: The curvature index *CI* field, defined only for points that have range measurements. Pixels for which the value of *CI* could not be computed are marked in black. Bottom left: The brightness edges obtained by Canny's edge detector. Bottom right: The reconstructed 3–D surface, with the estimated ground plane and the detected curb edge segment (marked in yellow).

considered a curb edge. The reasons for these types of error are discussed at length in the next section.

G. Shortcomings and Bottlenecks

Although the technique described in the previous subsections has shown excellent results in most cases, there are some open issues that will require more work. We discuss the two most relevant remaining problems in the following.

Stereo Artifacts. It is well known that, as a consequence of slight camera misalignment, a window–based correlation stereo may compute incorrect disparities in the case of strong edges that are almost parallel to the epipolar lines. An analytical description of this behavior can be found in [17]; it is similar in nature to the "aperture problem" of optical flow, which arises when computing the image velocity component orthogonal to the image gradient. Fig. 11 illustrates this phenomenon by way of experiments with a test pattern viewed by our stereo camera at different orientations. The test pattern was printed on a flat surface. The epipolar lines of the rectified images shown in the figure are horizontal; the re–projection calibration error was less than 0.2 pixels. When the edge pattern at the center of the image is vertical, the disparity is correctly computed; by rotating the stereo camera such that the lines in the pattern are almost horizontal, a rather large disparity error (positive or negative) is recorded.

This phenomenon may influence the performance of our curb detector, since strong brightness gradients on an otherwise flat surface (e.g., a line painted on the floor) may generate a non–null value of the curvature index *CI* and therefore a non–negligible value of the weight w assigned

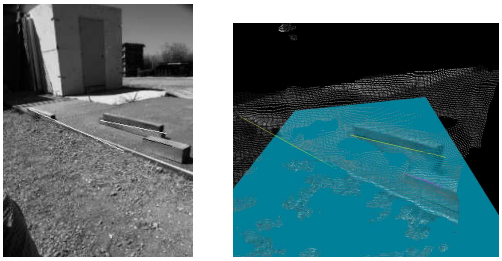

Fig. 6. Left: The left image in a stereo pair, with the detected curb lines (white: concave; black: convex). . Right: The reconstructed 3–D surface, with the estimated ground plane and the detected curb edge segments (yellow: concave; pink: convex).

to a pixel by (3). This may lead to errors in the determination of the segment endpoints (Fig. 9) and even to the incorrect detection of curb lines (Fig. 10).

Our current fix to this problem is to smooth the range data with a rather large kernel ($\sigma = 15$–21 pixels) before computing the curvature index, in the hope to reduce the influence of small range variations due to disparity estimation errors. However, a more comprehensive recovery strategy, based on a better understanding of the root of the problem (the disparity computation), is needed for more reliable results.

Peak Detection. An open problem with the system discussed so far is that any peak in the Hough transform that survives the non–maxima suppression test (Section II-D) is considered a candidate curb line – even when no curb is visible in the image! This calls for a criterion to assert with a certain degree of confidence when a curb is actually present in the image. We currently use the following heuristic. First, the median HT_m of the weighted Hough transform values is computed. Then, a peak is considered a candidate only if its value is larger

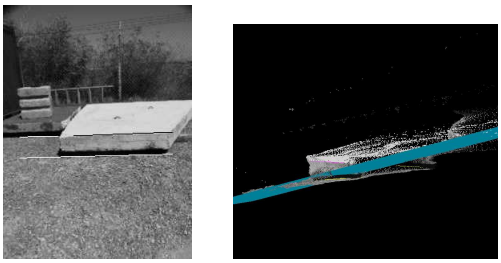

Fig. 7. Left: The left image in a stereo pair, with the detected curb lines (white: concave; black: convex). Right: The reconstructed 3–D surface (seen from behind), with the estimated ground plane and the detected curb edge segments (yellow: concave; pink: convex). Note that the estimated ground plane is incorrect, yet the curb edges are detected correctly.

Fig. 8. Left: The left image in a stereo pair, with the detected concave curb line. Right: The reconstructed 3–D surface (seen from behind), with the estimated ground plane and the detected curb edge segment (marked in yellow).

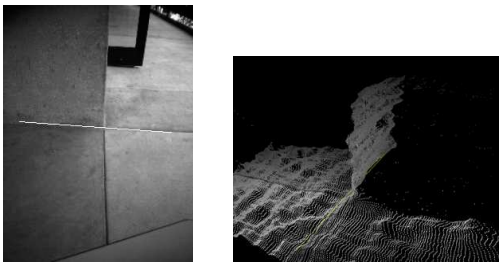

Fig. 9. Left: The left image in a stereo pair, with the detected concave curb line. Right: The reconstructed 3–D surface with the detected curb edge segment (marked in yellow). Note that the detected segment extends beyond the actual curb edge.

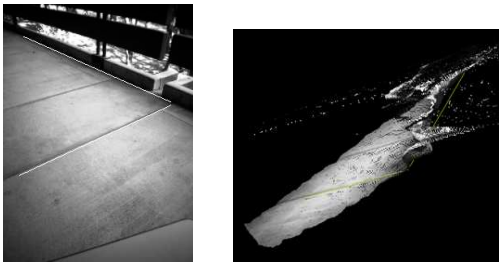

Fig. 10. Left: The left image in a stereo pair, with the detected concave curb lines. Right: The reconstructed 3–D surface with the detected curb edge segments (marked in yellow). Note that one of the two detected curb edges is incorrect.

than $S \cdot HT_m$, where S is a positive constant (e.g., $S = 5$). This procedure can be seen as a form of hypothesis testing. More precisely, we assume that in the "null" hypothesis (i.e., when there are no curbs in the scene), the values of the Hough transform tend to be distributed uniformly, while a heavy tail in the distribution would be observed when a curb is present.

This strategy has been only partially successful. Clearly, a more thorough investigation into the statistical properties of the weighted Hough transform for these types of images is needed for more satisfactory results.

III. Stairway Modeling

The curb detection and localization algorithm described in the previous section is the initial building block of our stairway modeling and localization procedure. Indeed, a stairway can be defined as a vertical succession of regular steps, with two sets of parallel convex and concave edges lying on parallel planes. Hence, if a number of such edges have been found, one may attempt to fit a plane through them, thus determining the stairway slope and step size.

Our algorithm proceeds as follows. First, a number of curb line segments are determined. The ground plane is not computed in this case (since very little of it may be visible anyway), and therefore relative height or slope criteria are not used to reject candidate lines. The line segments found are divided into the set of concave and of convex ones. For each such set, we run a variant of the LMedS algorithm to robustly fit a plane to the lines contained in the set. More precisely, we randomly select (with replacement) three lines, making sure that not all three lines are the same one. From each selected line, we randomly pick a point, making sure that the resulting three points are not collinear and that no two points are the same (in case two lines are the same one). Then, the distance (residual) of all points of all lines to the plane defined by such three points is computed[4], and the median residual is stored. After a number of iterations, the plane with the smallest median residual is chosen. This operation is performed on both line sets (concave and convex), and the plane with the smallest residual, $\bar{\Pi}$, is kept.

Once the tangent plane $\bar{\Pi}$ has been fitted, it would be desirable to find its intersections $\bar{l}_i$ with the planes Π_i defined in Section II-E. The reader is reminded that the plane Π_i contains the focal point and the line l_i in the image plane. The lines $\bar{l}_i$ thus contain the perspective projections onto $\bar{\Pi}$ of the points $\{p_j\}$ defined in Section II-E. Hence, besides being co-planar, the lines $\bar{l}_i$ are (ideally) parallel – even though the lines l_i in the image plane are, in general, *not* parallel, due to perspective. The knowledge of lines $\bar{l}_i$ may be very useful for the control of a stair–climbing robot [18]. However, this procedure does not eliminate the risk that a spurious line (e.g., a shadow) that is co–planar with the other edge lines be identified as an actual step line. To alleviate this risk, we can rely on the parallelism that is expected of the detected lines. A simple rejection procedure is outlined in the following. First, the angular differences, Δ_{ij} between the lines in each pair $(\bar{l}_i, \bar{l}_j)$ are computed. Then, for each fixed i, we compute

[4]Note that we use the actual 3–D points corresponding to image lines for this procedure, not the points in the line regressed by the procedure of Section II-E.

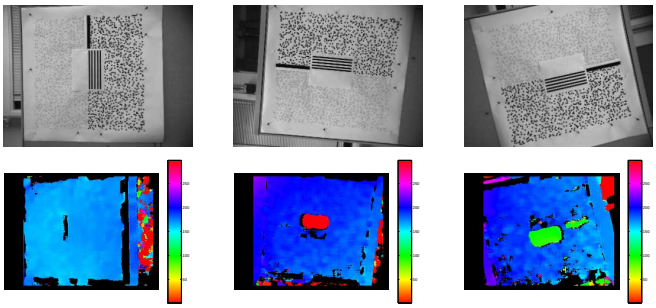

Fig. 11. Top row: The left image (rectified) of a stereo pair at different camera orientations. The printed pattern is almost perfectly flat. Bottom row: The computed disparity. Black pixels indicate points for which the disparity was not computed (due to lack of texture).

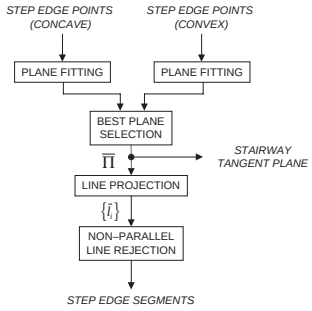

Fig. 12. A block scheme of the stairway modeling/localization algorithm.

the median of the values $\{\Delta_{ij}\}$. The resulting values are expected to be small for any line $\bar{l}_i$ that is almost parallel to most other lines, and high otherwise. The adaptive–scale outlier detection algorithm MSSE [1] is then used for rejection of these latter cases.

Examples of fitting plane $\bar{\Pi}$ and projected lines $\bar{l}_i$ (convex edges) for two different staircases are shown in Figs. 13–14. Note that non–parallel lines have been automatically rejected.

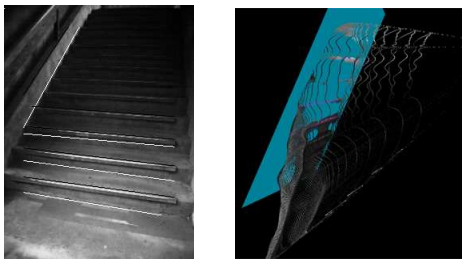

Fig. 13. Left: The left image in a stereo pair, with the detected step lines (white: concave; black: convex). Right: The 3–D stereo reconstruction (seen from behind), with the plane $\bar{\Pi}$ fitting the convex step lines, and the projected lines $\bar{l}_i$ (shown in purple).

IV. Summary and Conclusions

We have introduced new, robust algorithms for the detection and 3–D localization of curbs and of stairways. Our approach uses range data from stereo, correlating geometry with brightness edge information from the image. Implemented on a laptop computer, the overall system runs at about 4 Hz, with satisfactory results. Our system finds application in indoor and urban autonomous navigation, where curbs and stairways are important landmarks for world modeling.

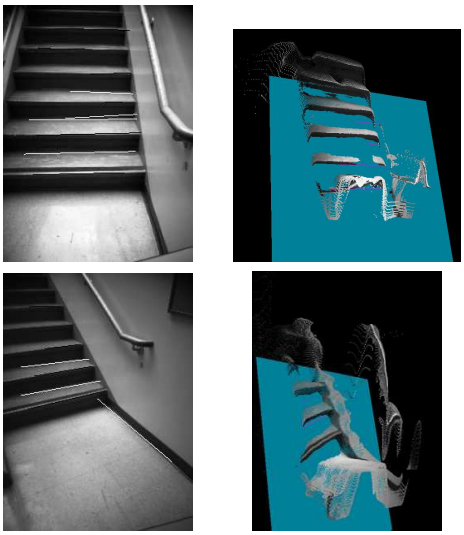

Fig. 14. Left: The left image in a stereo pair, with the detected step lines (white: concave; black: convex). Right: The 3–D stereo reconstruction, with the plane $\bar{\Pi}$ fitting the convex step lines, and the projected lines $\bar{l}_i$ (shown in purple). Note the stereo artifacts due to the textureless area at the bottom of the staircase.

References

[1] A. Bab–Hadiashar and D. Suter, "Robust Range Segmentation", *Proc. ICPR*, 1999.

[2] A. Broggi, "Robust Real–Time Lane and Road Detection in Critical Shadow Conditions", *Proc. IEEE Int. Symp on Comp. Vis.*, 1995.

[3] J. Crisman and C. Thorpe, "SCARF: A Color Vision System that Tracks Roads and Intersections", *IEEE Trans. Robotics and Autom.*, 9(1):49–58, Feb. 1993.

[4] E.D. Dickmans and B.D. Mysliwetz, "Recursive 3–D Road and Relative Ego–State Recognition", *IEEE Trans. PAMI*, 14:199–213, May 1992.

[5] D. Eberly, R. Gardner, B. Morse, S. Pizer and C. Scharlach, "Ridges for Image Analysis", *Journ. Math. Imaging and Vision* ,4(4):353–373, 1994.

[6] J. Forsberg, U. Larsson and A. Wernersson, "Mobile Robot Navigation Using the Range–Weighted Hough Transform", *IEEE Robotics & Automation Magazine*, March 1995.

[7] K. Lee, P. Meer, and A. Park, "Robust Adaptive Segmentation of Range Images", *Trans. IEEE PAMI*, 20(2), Feb. 1998.

[8] J. Miller and C. Stewart, "MUSE: Robust Surface Fitting Using Unbiased Scale Estimates", *Proc. IEEE CVPR'96*, San Francisco, 1996.

[9] P. Rousseeuw and A. Leroy, *Robust Regression and Outlier Detection*, Wiley, 1987.

[10] S. Se and M. Brady, "Vision–based Detection of Kerbs and Steps", *Proc. BMVC97*, 1997.

[11] S. Se and M. Brady, "Road Feature Detection and Estimation", *Machine Vision and Applications*, 14:157–65, 2003

[12] C. Thorpe et al., "Driving in Traffic: Short–Range Sensing for Urban Collision Avoidance", *Proc. SPIE: UGV Tech. IV*, Vol. 4715, April 2002.

[13] B. Tordoff and D.W. Murray, "Guided Sampling and Consensus for Motion Estimation"–, *Proc. ECCV*. May 2002

[14] P.H.S. Torr and C. Davidson, "IMPSAC: Synthesis of Importance Sampling and Random Sample Consensus", *IEEE Trans. PAMI*, 25:354–64, 2003.

[15] E. Trucco and A. Verri, *Introductory Techniques for 3–D Computer Vision*, Prentice–Hall, 1998.

[16] R. Turchetto and R. Manduchi, "Visual Curb Localization for Autonomous Navigation", *Proc. IEEE/RSJ IROS'03*, Las Vegas, October 2003.

[17] Y. Xiong and L. Matthies, "Error Analysis of a Real–Time Stereo System", *Proc. IEEE CVPR*, 1997.

[18] Y. Xiong and L. Matthies, "Vision–Guided Autonomous Stair Climbing", *Proc. ICRA*, 2000.