Visual Curb Localization for Autonomous Navigation

R. Turchetto, R. Manduchi
Department of Computer Engineering
University of California, Santa Cruz
Santa Cruz, CA 95060
{riccardo,manduchi}@soe.ucsc.edu

Abstract – Percept–referenced commanding is an attractive paradigm for autonomous navigation over long distances. Rather than relying on precise prior environment maps and self–localization, this approach uses high–level primitives that refer to environmental features. The assumption is that the sensing and processing system onboard the robot should be able to extract such features reliably. In this context, we present an algorithm for the visual–based localization of curbs in an urban scenario. This represents a basic sensing capability that would enable behaviors such as "follow this road keeping at a certain distance from its edge".

Our approach to curb localization relies on both photometry and range information (as obtained by stereopsis). Candidate image points are first selected using a combination of cues, and then input to a weighted Hough transform, which determines the line (or lines) that are most likely to belong to the curb's edge. Our experimental results show that, as long as the range data has acceptable quality, our system performs very robustly in real–world situations.

I. INTRODUCTION

Robots able to drive autonomously in urban environments would be of enormous help in a number of civilian and military applications. Unfortunately, current systems have only limited capabilities over long ranges. For example, visual–based tele–operation assumes that the sensor data can be transmitted reliably to the human operator. It is well known, however, that radio links are not reliable beyond line of sight, and the available bandwidth may be scarce. Systems relying on prior environment maps and self–localization do not require human operator intervention, and thus are not subject to the communication bottleneck. Self–localization is possible via differential GPS and dead–reckoning (including visual odometry [9]). Unfortunately, precise prior environment information is not an option in tactical scenarios, or when the available maps are outdated. Reception of GPS signals may be difficult in certain urban situations; and dead–reckoning is notoriously unreliable over very long distances.

Ideally, robots should not need precise, metric environment maps. Think at the way humans communicate spatial information in colloquial language – for example, when someone is in an unfamiliar area, and asks a local for directions to reach a specific place. Directions are hardly ever expressed in absolute metric terms (walk for five hundred thirty five feet toward NE, then turn toward SE, then walk for two hundred twenty feet, then stop and turn toward SW). Rather, directions are usually specified in terms of *landmarks* (keep walking on this sidewalk, turn right at the first traffic light, then look for the third building on your right). The assumption in the second case is that the interlocutor is familiar with the typical structure of a town, is able to perform basic actions (such as walking straight on a sidewalk), and can identify the specified landmarks (traffic light, individual buildings).

This simple example illustrates a few aspects of the *percept–referenced command* paradigm for robotics guidance. This approach may enable long–range autonomous navigation in the face of unreliable radio links, sketchy environment information, and inaccurate self–localization. Key to this strategy is in the ability of the robot's perceptual system to recognize the type of landmarks that are used to formulate commands.

This paper presents a solution to one of the several recognition tasks involved in percept–referenced commanding for missions in urban environments. In particular, we address an issue related the "road following" behavior, whereby the robot is instructed to follow a road, while keeping at a fixed distance from the road's edge. This is an elementary yet fundamental behavior that must be supported. The edge of most urban roads in the US is marked by a curb of a certain height (typically 4–7 inches). Thus, the robot's perceptual system needs to detect and precisely locate visible curbs in the scene (at a distance of possibly several meters).

Our curb localization system uses both geometric and photometric cues. In principle, given the simple structure of a curb, accurate range information (such as from a laser rangefinder) should be sufficient for detection. However, to detect curbs that are in unknown positions in front of the vehicle, two–axes lasers would be necessary. Such systems are slow, bulky and expensive. On the other hand, stereo systems are more affordable, can use miniaturized digital cameras, and provide frame–rate (or quasi–frame–rate) range maps using specialized or COTS hardware. An important advantage of stereo cameras is that images and range maps are "naturally" registered with each other (i.e., each range measurement corresponds to one pixel reading). This facilitates the implementation of pattern recognition algorithms that combine photometric and geometric information.

The range accuracy of stereo systems depends on a number of factors, including field of view, image

resolution, baseline, and photometric properties of the surfaces in the scene [16]. In our experiments we used JPL's real–time correlation–based system [16]. Although the range data produced by stereo is of sufficient quality for tasks such as obstacle detection [14], it does not always prove sufficiently accurate to reliably detect a 4–inches tall curb at a distance of a few meters, as required by our task. To improve the detection and localization performance, we complement geometric reasoning with image analysis. As a result of the sharp surface angles, an image area containing a curb typically has one or more pronounced edges. The presence of rectilinear edges does not, by itself, signal the presence of a curb. Strong brightness edges may be present due to a number of other reasons, including shadows or painted lines on the pavement. However, we show that by correlating edges with depth information, we are able to remove this kind of ambiguity, and obtain a more reliable result than by using depth or brightness alone.

A. Previous Work

There has been extensive work over the last fifteen years on visual lane and road detection for autonomous vehicles (see e.g. [2,1,3]). However, this work mostly considered problems such as lane marker and obstacle detection, and did not address specifically the detection of curbs.

Curb detection over short distances for safe driving has been demonstrated at CMU with a laser striper [13]. 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 [11]. 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. Our approach starts directly from range data, rather than from disparity values. Note that there exist a number of reasonably priced commercial off–the–shelf stereo systems that can compute range data in real time. Rather than the early–commitment approach of [11], 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 visual–based stair–climbing system proposed in [17] addresses a germane problem (a staircase may be seen as a succession of curbs). The challenge in [17], however, was not so much the detection of the staircase from the distance, but the control of the vehicle as it climbed upstairs.

II. ALGORITHM DESCRIPTION

As mentioned in the Introduction, our system uses both photometric and geometric information to localize a curb. The underlying principle is rather simple, and is outlined in the following.

The stereo system provides spatial information in the form $[x(i,j),y(i,j),z(i,j)]$, where (x,y,z) represent the location of a 3–D point in space (with respect to a coordinate frame moving with the robot) and (i,j) are the pixel coordinates of the projection of the point onto the image plane. Let's assume that the face of the curb is parallel to the Y axis, and that the edges of the curb are rectilinear and parallel. Under the weak perspective projection assumption [11], the projections of the two edges onto the image plane may be considered parallel, and the gradient of the "elevation" $y(i,j)$ (computed with respect to the axes I and J in the image) is orthogonal to such edges. At the same time, the projection of a curb's edge usually generates an image brightness edge. Thus, in the neighborhood of the projected curb's edge, the elevation gradient and the brightness' gradient are expected to both have high values and to be aligned. This observation is at the basis of our localization algorithm. Even though our geometric assumptions may hold true only approximately in practical cases, experiments have shown that our criterion based on the correlation of the gradients of brightness and elevation consistently produces good localization results.

Note in passing that color information is not very useful, in general, for curb detection (except when curbs are marked with different colors for parking regulation). This should result apparent by observing Figure 1; experiments with a statistical color classifier confirmed that color–based detection is bound to fail in most cases.

We describe the different components of our algorithm in the following.

A. Ground Plane Determination

The first step in our algorithm rotates the reference frame so that the edges of the curb are parallel to the *X*-*Z* plane. To this purpose, we approximate the road surface with a ground plane. A plane is clearly only a rough model for the road surface; however, this approximation works well enough for our purposes. After this change of basis, we only work on the elevation component $y(i,j)$ of the spatial information provided by stereo.

To estimate the ground plane, we use a robust procedure based on the Least Median of Squares (LMedS) algorithm [10]. An advantage of LMedS with respect to RANSAC (another standard procedure for robust fitting [7]) is that it requires no setting of thresholds or a-priori knowledge of the data statistics. LMedS, however, will fail when there are more than 50% outliers in the data. This may happen, for example, when the robot is facing a vehicle or a building, and only a small portion of the road surface is visible. To reduce this risk, we can use prior approximate knowledge of the orientation of the ground plane with respect to the reference frame. This information can be predicted from previous frames, assuming that the robot is not rocking or tilting too much. This knowledge may be used to reduce the number of points that are used for computing the LMedS fitting. For example, only points that, with respect to the reference frame have elevation within, say, ±1 meter from the predicted ground plane can be used for fitting. This allows us to dramatically reduce the number of outliers, and provides us with an acceptable initial guess. In addition, we only consider points that are within a certain distance (in our experiment: 5 meters) from the robot. This was suggested by two observations: first, points that are too far away have large range estimation errors; second, the planar assumption is usually acceptable only over limited size road patches.

Figure 1: The left image in a stereo pair.

Figure 2 shows the elevation $y(i,j)$ after ground plane normalization (i.e., change of basis) for the image of Figure 1. If the elevation values were reliable, and our geometric assumptions were valid, the field $y(i,j)$ alone could be used to detect curbs, by thresholding it at a

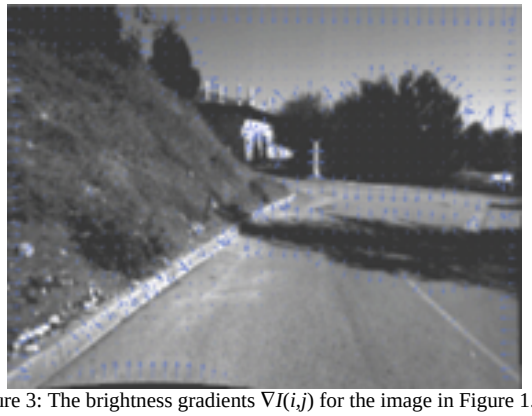


Figure 3: The brightness gradients $\nabla I(i,j)$ for the image in Figure 1.

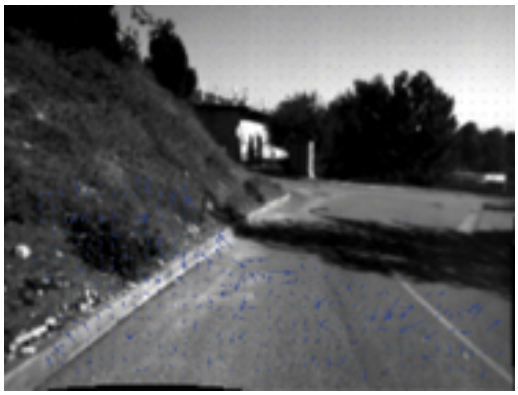


Figure 4: The elevation gradient $\nabla y(i,j)$ computed over the area highlighted in Figure 2.

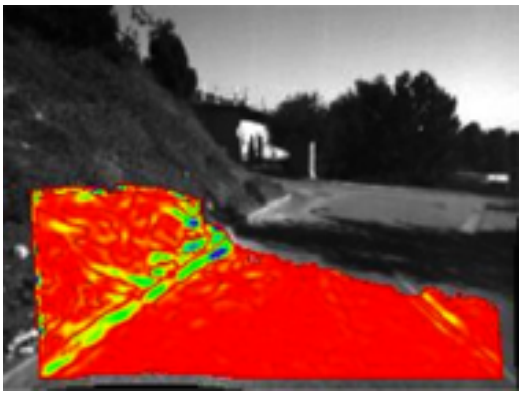


Figure 5: The correlation field $C(i,j) = |\nabla y(i,j) \bullet \nabla I(i,j)|$ (red = low values; blue = high values).

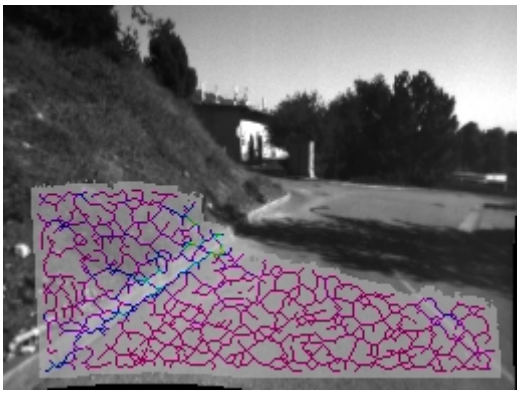


Figure 6: The ridges of $C(i,j)$. Each ridge point is colored according to the corresponding value of $C(i,j)$ (red = low values; blue = high values).

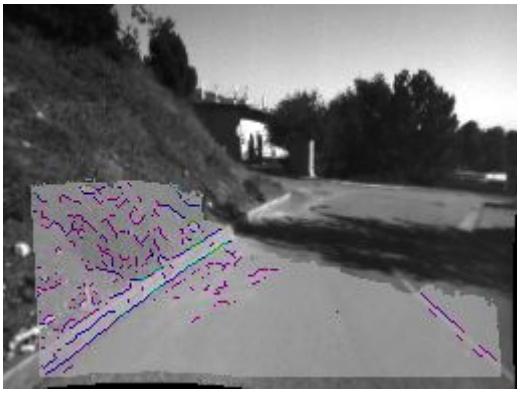


Figure 7: The brightness edges. Each edge point is colored according to the corresponding value of $C(i,j)$ (red = low values; blue = high values).

Figure 8: The localized curb edges (corresponding to the two highest values of the weighted Hough transform). Here as well as in the following figures the colored area correspond to the region used for curb localization (i.e., points that have elevation within ±0.5 meters from the ground plane, ad that are less than 7 meters away from the robot).

B. Correlating the Gradients

The next step in our algorithm is the computation of the gradients of the elevation field $y(i,j)$ and of the brightness field $I(i,j)$. To remove noise, we filtered brightness and elevation with a Gaussian kernel (in our tests we tried with standard deviations of 1-3 pixels; the correct value depends on the image resolution and on the noise characteristics). The two gradients, $\nabla y(i,j)$ and $\nabla I(i,j)$ are then correlated, following the assumption that, on a curb's face near its edge, $\nabla y(i,j)$ and $\nabla I(i,j)$ should be collinear and should both have large values. More precisely, we create a new scalar field $C(i,j)$ by taking the absolute value of the scalar product of $\nabla y(i,j)$ and $\nabla I(i,j)$ at each image point:

$$C(i,j) = |\nabla y(i,j) \cdot \nabla I(i,j)| \qquad (1)$$

where the dot represents scalar product. Points (i,j) where this *correlation field* $C(i,j)$ takes on low values have low likelihood to belong to the curb's edge, because one (or both) of the two gradients has small magnitude (e.g, a line painted on the road surface), or because the gradients are not aligned (e.g., a shadow cutting across the curb). However, a high value of $C(i,j)$ does not automatically signify the presence of a curb. In some situations, a very large brightness gradient (due, for example, to a sharp shadow edge) may occur in a small area where the elevation gradient is non–null, and the two gradients may accidentally align. This may happen, for example, in the case of a sloped surface outside the curb area, or due to range errors. (Note from Figure 4 that range gradient errors should be expected in practical cases.) Such isolated "outliers" may at times have a very high value of $C(i,j)$. We found that a simple limiter on the gradients helps reduce the unwanted effect of such occurrences. Still, looking at maxima of the correlation field is not enough to determine the curb's outline in the image. For example, some isolated points in the correlation field of Figure 5 have very high values although they do not belong to a curb. For robust localization, we should exploit the structural properties of curb profiles, which normally look like lines or line segments in the image. Our algorithm thus proceeds as follows. First, we extract candidate points in the image that have a high likelihood to belong to a curb's edge. Then, we look for lines in the image that best represent such candidate elements. We describe these two steps in the following.

C. Extracting Candidate Elements

We looked at two different options for the selection of candidate image elements. The first option considered the *ridges* of the correlation field. This strategy was inspired by [6], where the problem was to detect surfaces in a 3–D

"surface probability function". Ridges seem like a suitable choice here, because they reduce the risk of concentrating on isolated maxima.

Surface ridges can be computed by analyzing the Hessian matrix $\boldsymbol{H}$ of the field at each point. If k_1, k_2 and $\boldsymbol{v}_1$, $\boldsymbol{v}_2$ are the eigenvalues and the eigenvectors of $-\boldsymbol{H}$ at a given point (with $k_1 \geq k_2$), and $\boldsymbol{g}$ is the gradient of the field at the same point, then the point is a strong ridge point if

$$\boldsymbol{v}_1 \bullet \boldsymbol{g} = 0, \;\; k_1 > 0, \;\; \text{and } k_1 > |k_2| \qquad (2)$$

Ridge extraction thus requires (1) computing the first and second partial (and mixed) derivatives of $C(i,j)$, (2), computing the spectral decomposition of the Hessian, and (3) determining the zero–crossings of the function $\boldsymbol{v}_1(i,j) \bullet \boldsymbol{g}(i,j)$ over the image. An example of ridge extraction is shown in Figure 6, where each ridge point at (i,j) is colored according to the value of $C(i,j)$. Such values can be used as weights in the voting scheme to extract the curb's edges discussed in Section 2.4.

There are two problems with using ridges from the correlation field. First, note that the correlation surface is computed from gradient values. Suppose, for simplicity's sake, that we can neglect the contribution of elevation (e.g., the elevation gradient is constant). Then ridge computation involves the estimation of third–order derivatives of the image brightness, an operation that is notoriously sensitive to high–frequency noise. Second, errors in the elevation profile may reflect into localization errors for the ridge points. Elevation is obtained from stereo disparities, which may be inaccurate in some situations (for example, in the case of untextured surfaces).

Our second option for extracting candidate elements uses edges directly computed on the image brightness. In this way, (1) no higher–order derivatives need to be estimated, and (2) elevation errors do not contribute to localization errors. Correlation field values $C(i,j)$ are still used to weigh the edge points for the final voting procedure. Figure 7 shows the brightness edges (as extracted by a standard Canny edge detector [15]), colored according to the corresponding value of $C(i,j)$. Note that the edge detector already removed edges with low significance; the same suppression technique could be used to reduce the number of ridges in Figure 6.

Using edge elements from the brightness field may also help in cases where range information is missing due to failed disparity computation. A possibility we are currently exploring is to assign a low, "neutral" weight to those edge elements without an associated range value. Thus, these elements would still contribute, albeit with lower weight, to the final voting.

D. Line Voting Procedure

Once the candidate image elements have been identified, we can use a voting strategy to determine the line (or lines) that are most likely to correspond to the curb's edges. We use a weighted Hough transform for this purpose. The weighted Hough transform is not a new

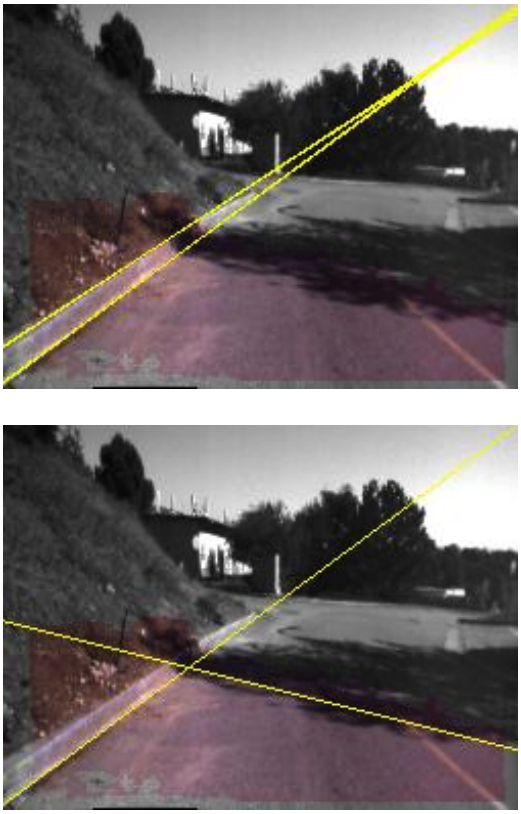


Figure 9: Curb localization using weighted Hough transform (upper image) and non–weighted transform (lower image). The lines correspond to the two highest values of the Hough transform.

concept in robotics; for example, it was used successfully in [5] to find the position and orientation of the walls relative to the robot, based on range data from a laser scanner. In the case of [5], a weight was used to compensate for the non–uniform spatial sampling rate. In our case, the weight of each ridge (or edge) point is equal to the associated value of $C(i,j)$. The resolution of the Hough transform determines the localization accuracy of the line in the image. In the experiments shown in this paper we have used a rather high accuracy (with an angular resolution of about 0.5 degrees and a distance resolution of approximately 1 pixel). However, if a lower accuracy is acceptable, the system works well also with a lower resolution (and consequently lower computational cost). To further reduce the computational weight associated with building the Hough transform, we may reject all candidate points with weight lower than a certain

threshold. In our experiments, we set this threshold to 1/20 of the maximum of $C(i,j)$. This procedure has the additional advantage of removing low–weight non–curb segments that in certain cases may be long enough to create local maxima in the transform. Examples of successful curb localization are shown in Figure 8, 9 and 10. In both cases, the Hough transform was computed over the brightness edge elements (second option of Section 2.3).

When there is only one curb visible in the image, the maximum of the Hough transform typically corresponds to a line representing one of the curb's edges. There are two more cases of interest that should be taken into account.

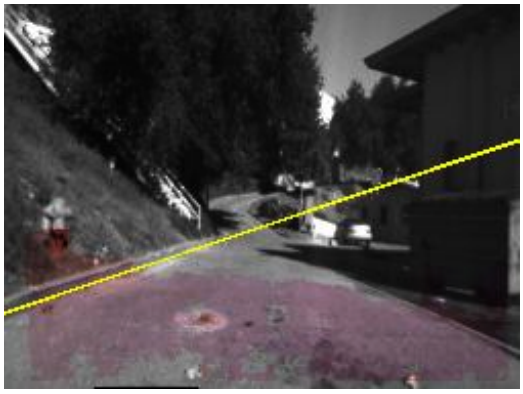
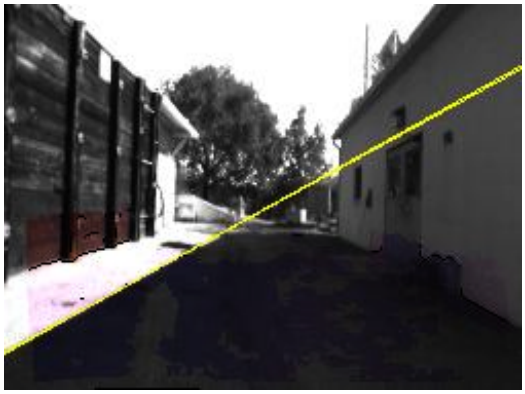


Figure 10: Another example of curb localization.

One case is when no curb is visible in the scene. In this situation, we expect the weighted Hough transform to be relatively flat, without a clear predominant maximum. Another case of interest is when there are more than one curb in the image. This will happen, for example, when the robot can see both sides of the road. In this case, the weighted Hough transform will present two distinct local maxima (or two distinct sets of local maxima). If these maxima all have comparable height, one may just select them all, and both curbs will be detected. However, in some cases (due, for example, to different illumination conditions) the two curbs may correspond to maxima with rather different height. A robust approach to deal with these two cases is still the object of current research.

Our choice of weights turned out to be very effective in reducing the risk of ambiguity. For example, in Figure 9 we show the result of detection with and without weighing the edge elements by the corresponding value of $C(i,j)$. In the second case, the detector (based only on brightness information) was fooled by an almost rectilinear shadow line.

One drawback of our system is that it relies on the accuracy of the range data. While a certain amount of range error usually does not create problems, there are



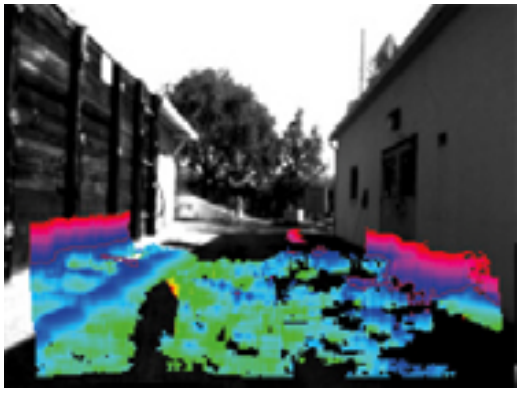


Figure 11: A case where our curb localization procedure fails due to bad range data. Note that the elevation map (lower image) shows an elevation "ridge" corresponding to the edge of the shadow. This is in fact a stereo matching artifact.

certain type of systematic errors that may lead to mis–localization. For example, in Figure 11 we show a case where our algorithm mistakenly detects the edge of a shadow. The problem in this case is that stereo generates an "elevation ridge" (as shown in the lower part of Figure 11), which clearly is a stereo matching artifact.

Some more examples on different sequences are shown in Figures 12 and 13, including both correct and incorrect localization results. In general, our algorithm gives mostly correct localization even in the presence of distractors (such us the white line painted on the pavement in Figure 13), as long as enough range values are available. Unfortunately, for situations such as in Figure 12 and 13, where the pavement has very little texture, range computation was often challenging. Another related problem is the computation of disparity for curb lines that are almost parallel to the epipolar lines of the stereo system.

III. CONCLUSIONS

We have presented an algorithm for localizing visible curbs in an image that makes use of photometric (image

Figure 12: Correct (above) and incorrect (below) curb localization.

brightness) and geometric (range data from stereo) information. The algorithm was tested on a number of real–world scenarios. Our experiments show that, as long as the range data is of acceptable quality, the system reliably localizes curbs in the image. We are extending the algorithm to also consider areas where range data is not available; in such cases, only information from brightness edges can be used. Our research is also addressing the general *detection* case, whereby the system must decide whether there are curbs in a given image and, if there is more than one curb, localize each one of them.

IV. ACKNOWLEDGMENTS

This work was supported by DARPA – MARS2020 Program, through subcontract 1080049-132209 from Carnegie Mellon University. The Machine Vision Group at JPL has kindly provided the stereo data used for the experiments.

V. REFERENCES

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

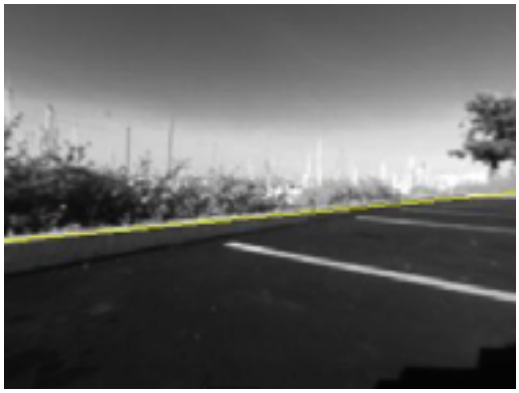


Figure 13: Correct curb localization in the face of distractors (white lines painted on the pavement).

[3] E.D. Dickmans and B.D. Mysliwetz, "Recursive 3–D Road and Relative Ego–State Recognition", *IEEE Trans. PAMI*, 14:199–213, May 1992.
[4] 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.
[5] J.Forsberg, U. Larsson and A. Wernersson, "Mobile Robot Navigation Using the Range–Weighted Hough Transform", *IEEE Robotics & Automation Magazine*, Mach 1995.
[6] A.E. Johnson and R. Manduchi, "Probabilistic 3–D Data fusion for Adaptive Resolution Surface Generation", *Proc. 1st Symp. 3PDVT*, Padova, Italy, 2002
[7] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2000.
[8] V.F. Leavers, *Shape Detection in Computer Vision Using the Hough Transform*, Springer–Verlag, 1992.
[9] L. Matthies, *Dynamic Stereo Vision*, Ph.D. Thesis, CMU, 1989.
[10] P. Rousseeuw and A. Leroy, *Robust Regression and Outlier Detection*, Wiley, 1987.
[11] S. Se and M. Brady, "Vision–based Detection of Kerbs and Steps", *Proc. BMVC'97*, 1997.
[12] S. Se and M. Brady, "Ground Plane Estimation, Error Analysis and Applications", *Robotics and Autonomous Systems*, 39 (2002), 59–71.
[13] C. Thorpe et al., "Driving in Traffic: Short–Range Sensing for Urban Collision Avoidance", *Proc. SPIE: UGV Tech. IV*, Vol. 4715, April 2002.
[14] A. Talukder, R. Manduchi. A. Rankin, and L. Matthies, "Fast and Reliable Obstacle Detection and Segmentation for Cross–Country Navigation", *Proc. Intell/ Vehicle Symp.*, 2002.
[15] E. Trucco and A. Verri, *Introductory Techniques for 3–D Computer Vision*, Prentice–Hall, 1998.
[16] Y. Xiong and L. Matthies, "Error Aalysis of a Real–Time Stereo System", *Proc. IEEE CVPR*, 1997.
[17] Y. Xiong and L. Matthies, "Vision–Guided Autonomous Stair Climbing", *Proc. ICRA 2000*.