Elevation-Based MRF Stereo Implemented in Real-Time on a GPU

V. Ivanchenko, H. Shen and J. Coughlan
Smith-Kettlewell Eye Research Institute
San Francisco, CA 94115

Abstract

We describe a novel framework for calculating dense, accurate elevation maps from stereo, in which the height of each point in the scene is estimated relative to the ground plane. The key to our framework's ability to estimate elevation accurately is an MRF formulation of stereo that directly represents elevation at each pixel instead of the usual disparity. By enforcing smoothness of elevation rather than disparity (using pairwise interactions in the MRF), the usual fronto-parallel bias is transformed into a horizontal (parallel to the ground) bias – a bias that is more appropriate for scenes characterized by a dominant ground plane viewed from an angle. This horizontal bias amounts to a more informative prior for such scenes, which results in more accurate surface reconstruction, with sub-pixel accuracy.

We apply this framework to the problem of finding small obstacles, such as curbs and other small deviations from the ground plane, a few meters in front of a vehicle (such as a wheelchair or robot) that are missed by standard real-time correlation stereo algorithms. We demonstrate a real-time implementation of our framework on a GPU (we have made the code publicly available), which processes a 640 x 480 stereo image pair in 160 ms using either our elevation model or a standard disparity-based model (with 32 elevation or disparity levels), and describe experimental results.

1. Introduction

Stereo is becoming increasingly useful as a real-time sensor for detecting important terrain features, such as obstacles, curbs and fall-offs, for vehicles such as wheelchairs (the particular application that motivated this paper) and robots. However, while standard real-time correlation window-based stereo algorithms are accurate enough to determine the dominant ground plane in a scene and to detect sufficiently large features at moderate range, the disparity maps estimated by such algorithms are often so noisy that smaller features may be lost in the noise – even at fairly close range. One solution to this problem is to use MRF stereo algorithms, which are significantly less noisy than correlation window-based stereo methods, and currently rank among the top-performing stereo algorithms [13]; while MRF algorithms are computationally intensive, they can now be implemented to run rapidly using off-the-shelf GPU hardware [16].

However, like most stereo algorithms, MRF stereo suffers from a fronto-parallel bias imposed by the disparity smoothness prior, typically expressed using pairwise (1st-order) interactions between disparities at neighboring pixels. This bias is inappropriate in the very situations that are crucial for terrain analysis, in which simpler algorithms fail: finding modest-sized (but important) deviations from the dominant ground plane, such as occur at curb edges viewed at a distance. In these cases, standard stereo algorithms easily establish that there are no *large* deviations from the plane, but limited precision and the presence of noise in the stereo correspondence limits the ability to detect small deviations (corresponding to sub-pixel disparity changes). Moreover, the fronto-parallel bias acts as an unrealistic prior since the dominant ground plane is viewed at an angle, and the bias can impair correct inference of the disparities on (and near) the ground plane.

To circumvent this problem, we propose a novel MRF formulation of stereo that directly represents elevation at each pixel instead of the usual disparity. By enforcing smoothness of elevation rather than disparity (using pairwise interactions in the MRF), the usual fronto-parallel bias is transformed into a horizontal (parallel to the ground plane) bias. In scenes dominated by a ground plane, this horizontal bias acts to *reduce noise in the estimation of disparities on the ground plane, which makes it easier to detect deviations from the ground plane* – i.e. obstacles such as curbs or stones.

We describe our algorithm and its real-time implementation on a GPU, and show experimental results demonstrating that it out-performs a conventional MRF stereo algorithm (which uses the standard disparity representation) in its ability to reconstruct ground plane surfaces and to detect obstacles.

2. Related Work

A vast range of research has been conducted on the use of stereo algorithms to detect and localize obstacles and other depth discontinuities; here we survey a handful of representative work in this area.

A variety of algorithms have been devised to exploit the dominant ground plane structure of typical street scenes. A classic example of this type of work is GOLD (Generic Obstacle and Lane Detection system) [1], which warps the left and right views so that all points lying on the ground plane are brought into correspondence (and thus simple image differences reveal points that lie off the ground plane). Another is V-disparity [9], which reduces the noise in reconstructing the dominant road surface by assuming a single dominant disparity in each row of the image. More recently, some research has addressed the need for improved disparity accuracy to find obstacles at long distances on flat surfaces, such as [11], which explicitly models non-planar road surfaces to improve obstacle detection, and [6], which uses sub-pixel disparity resolution and exploits a simple "gravitational" prior that enforces the tendency for disparities to decrease higher up in the image.

A related research theme is the development of algorithms to detect specific kinds of obstacles, such as curbs [10] and steps [12], which exploit the specific structure of such obstacles to minimize the deleterious effects of stereo noise. Some work in this category specifically addresses the detection of obstacles for wheelchair applications, using 3D representations such as an occupancy grid [14] or elevation map [7].

Most of the work described so far in this section uses standard stereo techniques to obtain disparity estimates and process this disparity information in novel ways. However, other research is concerned with making more fundamental improvements to stereo algorithms to better model typical 3D environments. In particular, this body of work seeks to improve upon the usual fronto-parallel bias imposed by standard correlation or MRF stereo algorithms [13], to reflect the fact that many surfaces in typical environments are planar but not fronto-parallel (i.e. slanted). One approach to capture such slanted surfaces is sweep stereo, which identifies dominant surface orientations in a scene and explores multiple depth candidates relative to each surface orientation; recent work [5] uses a GPU implementation to boost the speed of sweep stereo to real-time. A similar approach is the Manhattan stereo model [4] that solves for piecewise planar surfaces, aligned to dominant "Manhattan" directions in a scene, in an MRF framework. Finally, work on higher-order MRF stereo [15] improves upon the usual 1st-order smoothness prior, which enforces a fronto-parallel bias, with a 2nd-order prior, which enforces a more general co-linearity bias that is satisfied by planar structures at arbitrary slants.

Finally, recent work explores the speed-ups obtainable by implementing MRF stereo algorithms on GPUs using belief propagation (BP) [2], some of which even attains real-time performance [16].

Our work combines aspects of many of these different research threads. Motivated by the need to detect obstacles on a ground plane from a distance, we sought a way of improving upon the fronto-parallel bias imposed by standard MRF stereo. We accomplished this goal by using an elevation-based representation, which imposes a prior that is better suited to the environment, and which allows sub-pixel disparity resolution. By using belief propagation to perform inference with the model, we were able to implement the algorithm to run on a GPU in real time.

3. Motivation for Elevation Framework

Our main thesis is that for many stereo applications in which scenes are dominated by a ground plane, it is better to represent depth information at each pixel using *elevation* (height from the ground plane) rather than the standard disparity representation. The rationale for an elevation representation is two-fold: the representation is more compact and the smoothness prior expressed in this representation is more accurate. We explain these rationales in more detail below.

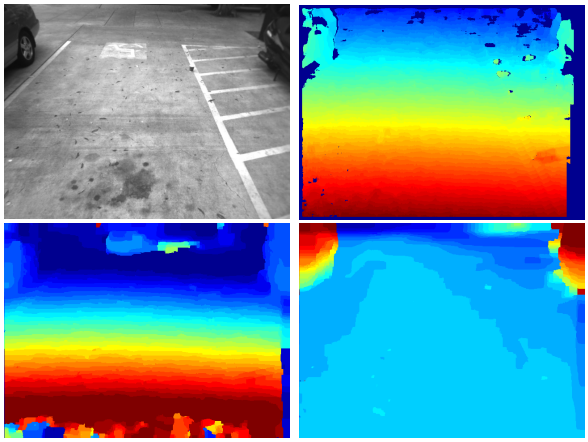

Figure 1. Top: (a) Input image showing typical ground plane scene. (b) Disparity map output using correlation window stereo. Bottom: (c) Disparity map from disparity-based stereo (disparity range is not wide enough in this implementation to cover top and bottom regions). (d) Elevation map from elevation-based stereo; few elevation levels are needed to represent ground plane.

First, the elevation representation is more compact for typical scenes (Fig. 1), in which most pixels have elevation close to zero, whereas the disparities of the ground plane vary widely from the bottom of the image to the top (requiring large disparity ranges of roughly 0-31 pixels). A small set of quantized elevation levels (e.g. 32 values ranging

across a 2-meter elevation span, from $-1m$ through $+1m$) suffice to finely resolve the ground plane geometry *throughout the image*, including positive and negative obstacles on the ground plane. However, the same number of disparity levels (e.g. disparities 0 through 31) might be used to cover the entire ground plane from the top to bottom of the image; note that many of the higher disparity levels are only relevant to the bottom part of the ground plane, leaving a limited number of levels to resolve geometric details in the upper part. Naturally, any fixed elevation range may be too narrow to encompass all obstacles (e.g. most trees are much higher than $1m$), but the focus of this paper is on resolving subtle geometric perturbations from the ground plane, and most standard stereo algorithms will detect large perturbations.

The second motivation for the elevation representation is that the smoothness prior expressed in this representation is more accurate for scenes dominated by a ground plane. The standard MRF stereo smoothness prior expresses the fact that disparities of neighboring pixels are similar. This disparity prior imposes a fronto-parallel bias, which is inappropriate in the situations we are interested in, when the camera line of sight is at an angle (e.g. roughly $45°$) to the ground, and so the ground plane disparity increases linearly (as a function of row coordinate) from the top to the bottom of the image. In such cases, a more appropriate prior would express the fact that elevations of neighboring pixels are similar.

Note that we can convert freely between the elevation and disparity representations. As shown in detail in Sec. 4, given the equation of the ground plane, the elevation at a given row and column in the image corresponds to a specific disparity (assuming knowledge of the stereo baseline and camera focal length). While the prior can be conveniently expressed either in terms of elevation or disparity, the likelihood model is naturally calculated in terms of disparity by evaluating the goodness of match between a pixel in the right image and the corresponding pixel in the left; for the elevation model, the likelihood evaluates all elevation hypotheses at a pixel by internally converting the hypotheses into disparities.

Finally, we note an important advantage of the elevation framework over an approach similar to the GOLD model (described in the previous section) that we originally considered, in which one of the two images is warped so that all points on the ground plane have zero disparity (and an explicit elevation representation is not needed for points on the ground plane). Unfortunately, with this approach, points on other planes parallel to the ground (i.e. planes with uniform, non-zero elevation) have *non-uniform* disparity (which can be proved from Eq. 7), so it would be difficult to use this approach to enforce a prior that favors smoothly varying (and often locally uniform) elevation.

3.1. Ground plane analysis

To better understand the benefit of the elevation prior, first consider a perfectly flat "reference" ground plane imaged under real-world imaging conditions (with camera noise). If the reference ground plane is defined by $\mathbf{R}\cdot\mathbf{n} = k$, where $\mathbf{R} = (X, Y, Z)$ are 3D camera-aligned coordinates (the positive Z-axis points along the camera line of sight, and the X axis is approximately parallel to the ground), then we can re-express this in terms of disparity as a function of pixel coordinates (u, v). (Here the center of the rectified image is $(u, v) = (0, 0)$; u increases with increasing column number and v increases with increasing row number.) First we recall the projection equations:

$$u = fX/Z, \quad v = fY/Z, \quad Z = fB/d \tag{1}$$

where d is disparity, f is the focal length and B is the camera baseline separation. Since $X = uZ/f$ and $Y = vZ/f$ we get $Z = fk/(\mathbf{n}\cdot(u, v, f))$. Then, since $Z = fB/d$ we obtain the disparity of the ground plane:

$$d(u, v) = fB/Z = B\mathbf{n}\cdot(u, v, f)/k \tag{2}$$

Note that the disparity is a linear function of the column and row coordinates. If the camera is held with the baseline horizontal (i.e. the X-axis is parallel to the ground), then $n_x = 0$ and the disparity of the ground plane is independent of pixel column and varies only from one row to the next.

If a stereo algorithm quantizes disparity in some way (e.g. integer values or some regular grid of values, as is typically used with a discrete-valued MRF stereo model), then the ideal estimated disparity field will jump from one disparity level to the next every few rows. A significant disadvantage of the quantized disparity representation is that *even a perfectly flat plane has disparity jumps if it is not fronto-parallel, and it may be difficult to distinguish such jumps from discontinuities due to obstacles on the ground plane.* As a consequence, *any attempt to smooth out the disparity estimates by penalizing disparity jumps will have the undesired side effect of also smoothing out true geometric discontinuities.*

Conversely, in the elevation representation, a perfectly flat plane will have a single elevation value across the image. Even if the plane isn't perfectly flat (e.g. a gradually sloping sidewalk) or the reference ground plane equation is slightly incorrect, the elevation will only change slowly across the plane. Given the quantization of elevation values, this means that *elevation jumps are rare on the ground plane.* Thus, *the elevation change due to a small obstacle is more likely to be detected.*

In summary, the elevation representation introduces less noise in its reconstruction of horizontal planes, and so it is more sensitive to small obstacles on these planes. Finally,

we note that the elevation representation is robust to small deviations from the horizontal (e.g. gently curved road surfaces).

4. Algorithm

We first describe the standard disparity-based MRF model [3], and then describe the proposed elevation-based MRF model, which is a simple variation of the first model.

4.1. Disparity-based MRF model

We are given the left and right grayscale images L and R, which are assumed rectified so that a pixel in one image matches a pixel in the same row in the other image. The unknown disparity field is represented by D, with $D_{\mathbf{r}}$ representing the disparity at pixel location $\mathbf{r}$. A particular disparity value $D_{\mathbf{r}}$, where $\mathbf{r} = (u, v)$, has the following interpretation: $(u + D_{\mathbf{r}}, v)$ in the left image corresponds to (u, v) in the right image.

We define a smoothness prior (i.e. binary potential in the MRF) on the disparity field D which enforces smoothness:

$$P(D) = \frac{1}{Z} e^{-\beta V(D)} \tag{3}$$

where Z is a normalizing constant ensuring that $P(D)$ sums to 1 over all possible values of D, β is a positive constant that controls the peakedness of the probability distribution, and $V(D) = \sum_{<\mathbf{rs}>} f(D_{\mathbf{r}}, D_{\mathbf{s}})$, where the sum is over all neighboring pairs of pixels $\mathbf{r}$ and $\mathbf{s}$. Here $f(D_{\mathbf{r}}, D_{\mathbf{s}})$ is an energy function that penalizes differences between disparities, and the particular form we use [3] is $f(D_{\mathbf{r}}, D_{\mathbf{s}}) = \min(|D_{\mathbf{r}} - D_{\mathbf{s}}|, \tau)$, which ensures that the penalty can be no larger than τ.

Next we define a likelihood function (i.e. unary potential in the MRF), which defines how the left and right images provide evidence supporting particular disparity values:

$$P(m|D) = \prod_{\mathbf{r}} P(m_{\mathbf{r}}(D_{\mathbf{r}})|D_{\mathbf{r}}) \tag{4}$$

where the product is over all pixels in the image, and m is the matching error across the entire image. Specifically, $m_{\mathbf{r}}(D_{\mathbf{r}})$ is the matching error between the left and right images assuming disparity $D_{\mathbf{r}}$, defined as

$$m_{\mathbf{r}}(D_{\mathbf{r}}) = |L(u + D_{\mathbf{r}}, v) - R(u, v)| \tag{5}$$

(again $\mathbf{r} = (u, v)$). If the value of $D_{\mathbf{r}}$ is fractional (i.e. sub-pixel resolution), then linear interpolation is used to estimate the value of $L(u + D_{\mathbf{r}}, v)$.

Finally, a simple model for the matching error is given by:

$$P(m_{\mathbf{r}}(D_{\mathbf{r}})|D_{\mathbf{r}}) = \frac{1}{Z'} e^{-\mu m_{\mathbf{r}}(D_{\mathbf{r}})} \tag{6}$$

4.2. Elevation-based MRF model

The elevation-based MRF model is the same as the disparity version above, except that the unknown disparity field is replaced by an unknown elevation field E. The form of the prior $P(E)$ remains the same, which penalizes elevation differences among neighboring pixels: $P(E) = \frac{1}{Z} e^{-\beta V(E)}$.

However, the likelihood is slightly different from before because it now evaluates each elevation hypothesis in terms of the corresponding disparity hypothesis. We now discuss precisely how this conversion is accomplished.

Given the equation of the ground plane, $\mathbf{R} \cdot \mathbf{n} = k$, where $\mathbf{n}$ points upward out of the ground, we define the elevation of any 3D point $\mathbf{R} = (X, Y, Z)$ to be $E(\mathbf{R}) = \mathbf{n} \cdot \mathbf{R} - k$. By definition, the elevation is zero on the ground plane, and points above the ground plane have positive elevation.

First we note that elevation can be simply computed as a function of disparity and pixel coordinate: $E(u, v, d) = \mathbf{r} \cdot \mathbf{n} - k = (Zu/f, Zv/f, Z) \cdot \mathbf{n} - k$ where $Z = fB/d$.

Next, we can solve for Z in terms of E, so $Z = \frac{(f)(E+k)}{(u,v,f)\cdot\mathbf{n}}$. Therefore we can convert elevation to disparity using the following equation:

$$d(u, v, E) = \frac{B}{E + k}(u, v, f) \cdot \mathbf{n} \tag{7}$$

The likelihood function has the same form as before: $P(m|E) = \prod_{\mathbf{r}} P(m_{\mathbf{r}}(E_{\mathbf{r}})|E_{\mathbf{r}})$, where $P(m_{\mathbf{r}}(E_{\mathbf{r}})|E_{\mathbf{r}}) = \frac{1}{Z'} e^{-\mu m_{\mathbf{r}}(E_{\mathbf{r}})}$. However, a crucial difference is that the matching error is evaluated by converting elevation into disparity, using Eq. 7:

$$m_{\mathbf{r}}(E_{\mathbf{r}}) = |L(u + d(u, v, E_{\mathbf{r}}), v) - R(u, v)| \tag{8}$$

Finally, we estimate the ground plane equation off-line, and assume that the definition changes only minimally over time since the stereo camera is fixed to the wheelchair platform it is mounted on. It would be straightforward to use a robust method for automatically determining the ground plane separately in each frame, but for the purposes of finding elevation discontinuities, our stereo algorithm is robust to small errors in the ground plane definition.

4.3. Inference and Implementation Details

We implemented the models as specified above but with a few added enhancements. First, the raw images were first smoothed with a Gaussian filter. Then, instead of defining the matching error solely in terms of the intensities of the left and right images, we added a second term to the matching error to measure the mismatch between the horizontal derivatives of the smoothed intensities. Since the horizontal derivative seemed to be a more reliable cue, we weighted the intensity term by 0.1 and the derivative term

by 0.9 before adding them. Second, we made the strength of the smoothness prior conditional on the strength of the image gradient: if the gradient between two adjacent pixels was above a threshold (suggesting the presence of an edge), then a weaker smoothness was used than if the gradient was below threshold. This is a standard procedure [13] exploiting the fact that depth discontinuities almost always occur at intensity discontinuities.

Inference in our model was performed using belief propagation (BP), specifically sum-product BP (which estimates the marginal probabilities at each pixel). (We also experimented with max-product BP, which gave similar results but took slightly more time to execute on the GPU.) In order to speed up convergence, and also to improve results (such as the ability to fill in low-texture regions with poor disparity evidence), we implemented the MRF stereo models in multiscale (using three scales), as described in [3]. To further improve fill-in behavior by discounting sufficiently noisy evidence, we modified the definition of the likelihood function as follows: if the evidence at any pixel is sufficiently ambiguous, then we "flatten" the likelihood function at that pixel by assigning equal likelihood to all elevation (or disparity) values. More precisely, we flatten the likelihood at a pixel if the top likelihood score for a particular elevation (or disparity) state is less than 10% of the sum of the scores for all states at that pixel.

We set the model parameters by trial and error, separately for each model, so as to make each model reconstruct both ground planes *and obstacles* as cleanly as possible. In the future, automatic learning procedures can be used (including unsupervised techniques such as [17]) instead to improve the model parameters.

Finally, we note that the code implementations for the elevation and disparity models are very similar: the only fundamental difference is in the unary potential calculation embodied in the changes between Eq. 5 and Eq. 8.

5. GPU Implementation

As far as we know, the fastest existing MRF stereo GPU implementation is [16], which processes 320x240 images at 16 fps (62.5 ms/frame) with 16 disparity levels. By contrast, our algorithm processes each 640x480 image in 160 ms with 32 elevation/disparity levels. Even if the number of disparity levels in the two implementations were the same, ours would be faster on a per-pixel basis (since the computation time scales linearly with the number of pixels) – but our performance is better still given that we process twice as many elevation/disparity levels. In addition, we have made our code freely available on our website [8] and in this section we describe important implementation details.

We tested our implementation of stereo algorithm on two different GPUs (GeForce 9800 GTX, GeForce GTX 260) with compute capabilities 1.1 and 1.3, respectively (the speeds reported above were using the second, faster of the two GPUs). A very serious processing bottleneck of GPU implementation was storing and updating BP messages that are kept in global memory: the GPU's global memory is much slower than constant, shared, register or texture memories. To speed up the access to the global memory we had to fulfill the so-called "coalescing" requirement, which essentially means that the innermost thread index should correspond to adjacent memory locations (e.g. levels of disparity). This requirement defines the division of labor into threads and blocks as shown in Fig. 2.

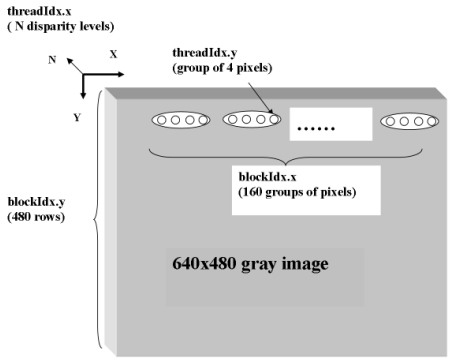

Figure 2. Structure of threads and blocks that satisfies coalescing requirement. The innermost thread index (threadIdx.x) accesses consecutive locations in global memory (N levels of disparity). The other dimensions were chosen based on the number of registers and shared memory consumed by a kernel so as to maximize its processor occupancy.

To summarize, kernel invocations used two dimensional blocks that consisted of threads that indexed disparity levels (innermost dimension) as well as several row pixels (another dimension). The grid structure was also 2D and used blocks that indexed groups of pixels in the same row and blocks that indexed rows. The number of pixels in a group varied from 4 to 16 depending on the kernel's memory consumption so as to maximize its processor occupancy.

An important aspect of BP implementation is the message update schedule. Synchronous updates typically result in a slow convergence since there is only limited propagation during the update. In comparison, during an asynchronous update, messages travel back and forth across the whole image on a single iteration. To implement such an update on the parallel architecture one has to ensure that only one row/column is processed at a time (i.e. in parallel) for horizontal/vertical directions of update. The challenge of doing this arises from the fact that the order of block loading into GPU is undefined, while using explicit loops has a huge kernel/block processing overhead. However, we were able to figure out the pattern of block loading for each compute capability and thus optimize the performance with respect to the update schedule.

Another important feature was the use of textures on the GPU, which speeds up access to global memory since textures are well cached when accessed locally in 2D. (The main disadvantage of texture memory is that it only provides read-only access.) In our implementation, textures were used to store all image data and binary (smoothness) potentials. We did not use GPU-native texture interpolation for sub-pixel disparity access since it was slower than the one we implemented in our code. Because of their vast size, unary potentials could not be stored as texture and were put into global memory instead.

The following routines were implemented on the GPU: image smoothing, image gradient, creating unary potentials, message initialization, belief propagation, belief calculation, and finding the the most likely elevation/disparity state (the "winners") at each pixel. The following operations were included in the timing that was 160ms per 640x480 frame (compute capability 1.3): loading a stereo pair from the hard drive, image smoothing and gradient calculation, creating unary potentials for all three scales, message initialization, running BP for two iterations on each scale, calculating beliefs, finding the winners, and copying them to the host.

6. Experimental Results

Since our elevation-based stereo model is intended for scenes dominated by ground planes viewed from an angle, we did not attempt to evaluate the algorithm on a standard dataset such as the Middlebury dataset (which does not contain those sorts of scenes). Instead, we compare the elevation-based model with the disparity model on typical outdoor scenes of interest, captured by a Point Grey Bumblebee 2 stereo camera (grayscale images at 640 x 480) mounted on a wheelchair ([7]). Both models use the same type of disparity evidence but employ different representations and different priors; while both models can be improved in various ways, the *relative* performance of the two models in their current form should reflect how much of an improvement is due to the elevation representation.

Since no *metric* ground truth range or disparity data is available for these scenes, we decided to use a simpler form of ground truth information: knowledge of which paved regions in the scene can be classified as locally flat (neglecting minor elevation discontinuities due to leaves or other litter on the surface) or as non-flat (containing a discontinuity such as a curb or obstacle, which is either clearly visible or was known to the authors when we captured the images).

We conducted two experiments using this ground truth information, one to evaluate the noise in reconstructing the ground plane, and a second to evaluate the ability of the models to discriminate flat regions from discontinuous regions. Discontinuities arise from either "positive" obstacles that protrude from the ground, such as a rock on the pavement, or "negative" obstacles, such as curb boundaries with elevation drops.

Before describing the experiments in detail, we first describe measures we took to equate the two stereo models as closely as possible. We chose the elevation and disparity ranges such that (a) the ranges were adequate for reconstructing the ground plane and any obstacles in a region of interest in the image and (b) the disparity resolution was similar for both models, i.e. the difference in adjacent disparity levels (corresponding to the consecutive elevation states in the elevation model) was approximately equal to the difference in disparity levels in the disparity model. We chose the elevation levels to range from $-0.4m$ to $+0.8m$, equally spaced over $N = 32$ levels (the maximum number of levels that our GPU stereo implementation currently handles); this implies that the range of disparities in the disparity model was from 11 to 29 pixels, which was adequate for capturing the ground plane in most of the image except for some rows near the top and bottom. (Rows outside that range would require disparities too low or high to fit in the allowed range, thus explaining the noisy regions at the top and bottom of Fig. 1(c).)

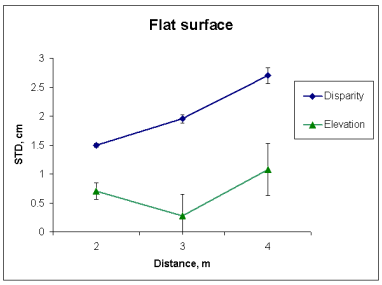

Figure 3. Standard deviation of elevations estimated by the two stereo models (elevation and disparity) on regions known to be flat. The standard deviations (i.e. noise levels) are lower for the elevation model than the disparity model, and these standard deviations tend to increase with distance to the camera.

The first experiment, which evaluates the noise in reconstructing the ground plane, provides empirical evidence for the claims made in Sec. 3.1. Regions of three images of sidewalk scenes were manually identified as being locally flat (i.e. no depth discontinuities inside), and were categorized according to their distance to the camera. The standard deviations of the elevation inside these regions were estimated using the elevation-based stereo model. Similarly, the disparity-based stereo model was run on the same images, and the resulting disparity maps were converted to elevation estimates using the definition of the reference ground plane; these estimates were used separately to estimate the average and standard deviation of the elevation. The results are shown in Fig. 3, which shows that the standard deviations (i.e. noise levels) are lower for the elevation model than the disparity model, and that these standard de-

viations tend to increase with distance to the camera.

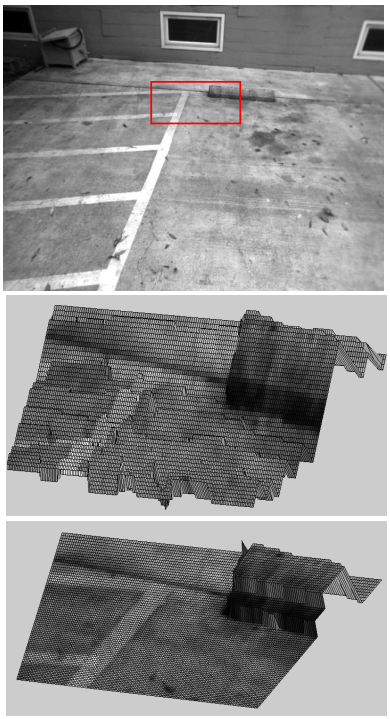

Figure 4. 3D reconstructions of a positive obstacle (the concrete parking block) from the two models. Top: original image, with region of interest outlined in red. Middle: reconstruction from disparity model. Bottom: reconstruction from elevation model. The obstacle is more clearly discriminable from the flat background in the elevation model.

Naturally, the noise in estimating the ground plane with the elevation model could easily be decreased merely by increasing the strength of the prior. However, Fig. 4 shows 3D reconstructions due to each model, suggesting that the elevation model is better at reconstructing both flat regions *and* depth discontinuities. To explore this difference further, we conducted a second experiment to investigate the ability of both stereo models to discriminate between flat surfaces and discontinuous ones. We considered four different obstacles: three positive obstacles (a rock, drinking cup and a concrete parking block) and one negative obstacle (a curb viewed from the sidewalk), viewed from a range of distances. The positive obstacles appeared in a total of 15 images, and there were 8 other images with the negative obstacle, and all obstacles were recorded at a distance of $4m$ from the camera.

We then defined a local measure of elevation discontinuity, and measured the distribution of this measure on flat and non-flat regions of the image using an ROC curve. We refer to this local measure of elevation discontinuity as a *score*, which is defined as follows. The score at any pixel location is defined in terms of the 50 x 50 pixel patch centered at that location, and equals the 95th percentile elevation value minus the 5th percentile value in the patch. The score has units of elevation difference (in meters), and is a more robust version of a simpler scoring function equal to the maximum minus minimum elevation value in a patch. To permit as fair a comparison as possible between the elevation and disparity models, the output of the disparity model is converted into elevation values before being evaluated by the score function.

For each image in our dataset, a row in the image was chosen such that when a pixel patch centered on the row is scanned from left to right (each time moving the patch 25 pixels to the right), some of the patch locations would intercept an obstacle. The patch locations with obstacles in them were manually classified as non-flat regions, while the other regions were classified as flat regions, thus establishing ground truth for the experiment. Flat patches tended to have lower scores than non-flat patches; we quantified this trend using an ROC curve in which the false positive and true positive rates are determined by sweeping a score threshold.

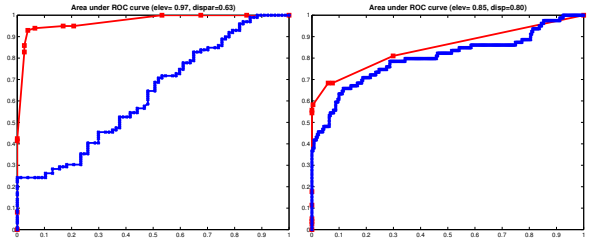

Figure 5. ROC curves showing the discriminability of flat vs. non-flat surfaces according to a simple elevation discontinuity measure. The red curves correspond to data generated by the elevation model, and the blue curves to data generated by the disparity model; the left figure is for negative obstacle data, and the right figure is for positive obstacles. In both cases, the AUC is higher for the elevation model than the disparity model.

Separate ROC curves are shown (Fig. 5) for positive and negative obstacles, with a total of 176 patches for positive obstacles and 330 patches for negative obstacles used to compute each ROC curve. For both types of obstacles, the AUC (area under the ROC curve) was higher for the elevation model than the disparity model.

7. Conclusion

We have described a novel MRF stereo framework based on explicitly modeling elevations, rather than disparities, which is useful for scenes of typical man-made environments with dominant ground plane structures that are viewed from an angle. The main rationale for the elevation representation is that a smoothness prior on elevations is more appropriate for these environments than one based on disparities, since the latter imposes an unnatural fronto-parallel plane bias. An additional important advantage of

the elevation representation is that it is more compact for many scenes, in the sense that a small range of elevations will be appropriate for resolving structure on and near the entire ground plane surface, whereas many disparity levels are required to accommodate near and far points on the ground plane.

Our GPU implementation (both the elevation-based MRF and disparity-based MRF code are freely available at [8]) achieves real-time performance: 160 ms per stereo pair at 640 x 480 pixel resolution (and $N = 32$ elevation or disparity levels). Experimental results demonstrate that our approach reconstructs ground plane structures with less noise than the disparity-based model, which means that small deviations (obstacles) are easier to resolve.

In the future we will explore several possible enhancements to our framework. We will consider using belief probabilities (estimated by BP) to estimate average elevation (or disparity) with a resolution finer than the quantized levels, and may use pairwise beliefs to estimate the presence of discontinuities between adjacent pixels. We will also explore searching for specific obstacle structures using specialized measures (e.g. ridge strength as in [10] to find curbs). Finally, we will investigate the use of our algorithm in the application that originally motivated it, which is to detect obstacles and other terrain hazards for blind wheelchair users.

8. Acknowledgments

The authors were supported by the National Science Foundation (grant no. IIS0415310).

References

[1] M. Bertozzi, S. Member, A. Broggi, and A. Member. Gold: A parallel real-time stereo vision system for generic obstacle and lane detection. *IEEE Transactions on Image Processing*, 7:62–81, 1998.

[2] A. Brunton, C. Shu, and G. Roth. Belief propagation on the gpu for stereo vision. In *Canadian Conference on Computer and Robot Vision (CRV06)*, pages 76–76, 2006.

[3] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient belief propagation for early vision. *Int. J. Comput. Vision*, 70(1):41–54, 2006.

[4] Y. Furukawa, B. Curless, S. Seitz, and R. Szeliski. Manhattan-world stereo. In *CVPR09*, 2009.

[5] D. Gallup, J. Frahm, P. Mordohai, Q. Yang, and M. Pollefeys. Real-time plane-sweeping stereo with multiple sweeping directions. In *CVPR07*, pages 1–8, 2007.

[6] S. K. Gehrig and U. Franke. Improving stereo sub-pixel accuracy for long range stereo. In *ICCV07*, 2007.

[7] V. Ivanchenko, J. Coughlan, W. Gerrey, and H. Shen. Computer vision-based clear path guidance for blind wheelchair users. In *Assets '08: Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility*, pages 291–292, New York, NY, USA, 2008. ACM.

[8] V. Ivanchenko, H. Shen, and J. Coughlan. Gpu stereo code for download. *www.ski.org/Rehab/Coughlan_lab/GPUstereo*.

[9] R. Labayrade, D. Aubert, and J.-P. Tarel. Real time obstacle detection on non flat road geometry through 'v-disparity' representation. In *Proceedings of IEEE Intelligent Vehicle Symposium*, volume 2, pages 646–651, Versailles, France, 2002.

[10] X. Lu and R. Manduchi. Detection and localization of curbs and stairways using stereo vision. In *IEEE International Conference on Robotics and Automation (ICRA '05*, 2005.

[11] S. Nedevschi, R. Danescu, D. Frentiu, T. Marita, F. Oniga, C. Pocol, T. Graf, and R. Schmidt. High accuracy stereovision approach for obstacle detection on non-planar roads. In *in IEEE Inteligent Engineering Systems (INES*, pages 211–216, 2004.

[12] V. Pradeep, G. Medioni, and J. Weiland. Piecewise planar modeling for step detection using stereo vision. In *CVAVI08, a satellite of ECCV08*, 2008.

[13] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47:7–42, 2001.

[14] B. J. H. J. Viswanathan, P. and A. Mihailidis. A comparison of stereovision and infrared as sensors for an anti-collision powered wheelchair for older adults with cognitive impairments. In *Proc. of the 2nd International Conference on Technology and Aging (ICTA)*, 2007.

[15] O. Woodford, P. Torr, I. Reid, and A. Fitzgibbon. Global stereo reconstruction under second order smoothness priors. In *CVPR08*, 2008.

[16] Q. Yang, L. Wang, and R. Yang. Real-time global stereo matching using hierarchical belief propagation. In *BMVC06*, page III:989, 2006.

[17] L. Zhang and S. M. Seitz. Estimating optimal parameters for mrf stereo from a single image pair. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(2):331–342, 2007.