Attenuating Stereo Pixel-Locking via Affine Window Adaptation

Andrew N. Stein
Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania
anstein@cmu.edu

Andrés Huertas and Larry Matthies
Computer Vision Group
Jet Propulsion Laboratory
Pasadena, California
{Andres.Huertas, Larry.Matthies}@jpl.nasa.gov

***Abstract*— For real-time stereo vision systems, the standard method for estimating sub-pixel stereo disparity given an initial integer disparity map involves fitting parabolas to a matching cost function aggregated over rectangular windows. This results in a phenomenon known as *pixel-locking*, which produces artificially-peaked histograms of sub-pixel disparity. These peaks correspond to the introduction of erroneous ripples or waves in the 3D reconstruction of truly flat surfaces. Since stereo vision is a common input modality for autonomous vehicles, these inaccuracies can pose a problem for safe, reliable navigation. This paper proposes a new method for sub-pixel stereo disparity estimation, based on ideas from Lucas-Kanade tracking and optical flow, which substantially reduces the pixel-locking effect. In addition, it has the ability to correct much larger initial disparity errors than previous approaches and is more general as it applies not only to the ground plane. We demonstrate the method on synthetic imagery as well as real stereo data from an autonomous outdoor vehicle.**

I. Introduction

Real-time stereo vision has proven to be a viable, cost-effective method for acquiring range data necessary for autonomous navigation. Range is determined from an estimated disparity map, which is the set of correspondences between pixels in the left and right images. More precise disparities produce more accurate range data, which will of course result in safer, more reliable navigation and obstacle detection. Since most, if not all, real-time stereo algorithms start by estimating disparities at the integer level, refining those estimates to sub-pixel accuracy is usually necessary to achieve highly accurate range estimates, which depend quadratically on disparity. This is particularly true when either the imagery is low-resolution or the stereo setup has a narrow baseline (small distance between the cameras). In either case, the total disparity range is small, meaning the actual 3D range is heavily quantized.

Typically, sub-pixel disparity is estimated by fitting parabolas to the cost function used for matching data between the left and right images. The analytical minimum cost can then be determined and the corresponding fractional offset used to adjust the initial integer disparity. This is highly efficient, but has widely been observed to result in "pixel-locking": a disproportionate number of sub-pixel disparity estimates around the initial integer disparities. This phenomenon is depicted in Figure 1. For a planar structure, the histogram of sub-pixel disparities should be perfectly uniform, as shown in the top plot. Starting with initial integer disparities, simple parabola fitting results in the obviously-peaked sub-pixel disparity histogram shown in the middle. The method described in this paper is capable of producing the much flatter histogram at the bottom. If the planar structure is the ground plane, for example, parabola fitting would produce an artificially-rippled 3D reconstruction of the ground while our approach would produce a more accurate flat surface, potentially leading to improved navigation and planning capability.

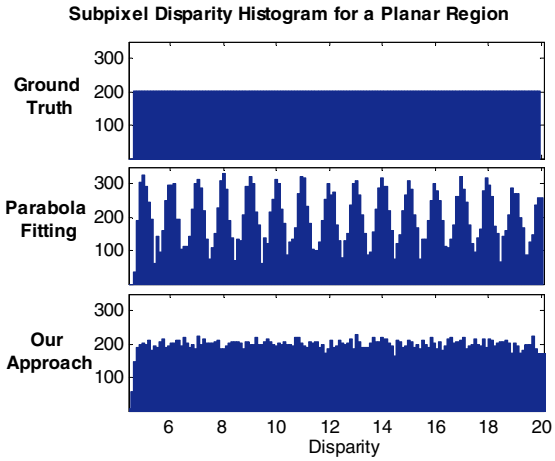

Fig. 1. Example histograms of sub-pixel disparities for a planar region, illustrating the "pixel-locking" effect. Using standard parabola fitting artificial peaks are clearly visible in the histogram, while our approach produces the desired nearly-uniform distribution, closely matching ground truth.

It is important to note that our method is applicable not only to estimation of a single ground plane. The reconstruction of *any* locally-planar structure in the scene will be improved. This differs from several other approaches ([1], [2], [3], [4]), and has particular application in man-made environments (particularly indoor), where many planar structures with varying orientations often exist throughout the scene. In addition, for very rough terrain where there may be significant bumps and ditches that deviate from an assumption of a single ground plane, other methods may not produce accurate reconstructions of these important obstacles. Our approach only makes assumptions of *local* planarity, and should be able to handle such conditions.

As will be described, our method refines initial integer

disparities by using local estimation techniques in the spirit of classical optical flow or the Lucas-Kanade (and Tomasi) tracker ([5], [6]). The method utilizes the original imagery (instead of an already-computed cost function) and adaptable affine windows to mitigate foreshortening effects and produce more accurate sub-pixel disparity estimates. In addition, it has the ability to correct for much larger errors in the initial integer disparity estimates than existing methods. A more detailed description of our technique, its relation to existing work, and results on synthetic and real imagery are provided in the remaining sections.

II. Computing Sub-Pixel Disparities

This section details the approach we follow, first providing notation and a mathematical formulation of the problem. Then, determination of the initial integer disparity map is described along with the typical parabola-fitting approach and an existing improvement method, followed by details on our approach.

Without loss of generality, let us assume our stereo pair is constructed such that we have left and right images, $I_L(x, y)$ and $I_R(x, y)$, and that the images have been rectified such that corresponding pixels lie on the same horizontal scanline in both images. Treating the left image as the reference image, we seek to find the best disparity map, $d(x, y)$, that matches a pixel (x, y) in I_L to its corresponding location $\big(x-d(x, y), y\big)$ in I_R so as to minimize some matching function $match(\cdot, \cdot)$. In order to reduce ambiguity when finding the minimum matching cost along a given scanline, the matching cost is aggregated over a window around (x, y), designated by $W_{(x,y)}$:

$$d(x, y) = \arg\min_{d} \sum_{(i,j)\in W_{(x,y)}} match\big(I_L(i, j), \quad I_R(i - d, j)\big)$$

For the remainder of this paper, we will consider the desired disparities in two parts: a coarse integer part, $d_{int} \in \{0, 1, 2, ..., D_{max}\}$, and a small fractional offset, $d_{off} \in \mathcal{R}$, such that the best estimate for the total disparity at each pixel is the sum of the best estimates of each of these parts:

$$d = d_{int} + d_{off}$$

A. Initial Integer Disparities

To compute the initial integer disparity at each pixel, we must compute the matching cost between $I_L(x, y)$ and $I_R(x - d_{int}, y)$ for each integer value of d_{int} and choose the one which results in the minimum cost. The matching cost computed at each pixel and every disparity is often referred to as the Disparity Space Image, $DSI(x, y, d)$. Thus, the initial integer disparity map is related to the DSI by:

$$d_{int}(x, y) = \arg\min_{d} DSI(x, y, d).$$

B. Parabola Fitting and Pixel Locking

A selected integer disparity $d_{int}(x, y)$ is, by definition, a local minimum of $DSI(x, y, d)$ along the d-dimension. We could therefore fit a parabola to the three cost values centered at $DSI(x, y, d_{int}(x, y))$ and find the sub-pixel location of that minimum. If we let $C(s) = DSI(x, y, d_{int}(x, y) + s)$, the offset from center of the analytical minimum of this parabola can easily be determined as

$$d_{off}(x, y) = \frac{C(-1) - C(1)}{2C(-1) - 4C(0) + 2C(1)}.$$

While obviously simple to implement, this method results in the pixel-locking effect discussed above and depicted in Figure 1. Shimizu and Okutomi [7] study this effect in detail and analytically derive a function describing the error in sub-pixel disparity estimation. Observing the error function's symmetric properties and its period of one pixel width, they attempt to cancel the theoretical errors as follows (see [7] for more details):

1) Compute fractional offsets to initial integer disparities using parabola fitting, as described above.
2) Effectively recompute the DSI using image intensity values interpolated at half-pixel locations in the left image (either 0.5 pixels to the left or right of the original sampling, depending on the direction of the initial offsets from step 1).
3) Recompute fractional offsets using parabolas fit to the DSI created in step 2 from the interpolated image data. Compensate the results by ± 0.5 to account for the interpolation.
4) Average the initial offsets from step 1 with those found in step 3.

The analysis in [7] also predicts a significant (approximately five-fold) attenuation of pixel-locking error simply by employing squared differences instead of absolute differences for the matching cost function. They also observe this effect experimentally, as do we. Noting that many existing real-time stereo approaches utilize absolute differences for their improved robustness on real-world data, we nevertheless use squared differences for the matching function in the remainder of this paper for consistency and assurance that we are not exacerbating unnecessarily the pixel-locking problem we seek to ameliorate[1].

C. Our Approach: Leveraging Lucas-Kanade

Unless all pixels in the window W over which matching cost is aggregated actually have the same disparity (corresponding to a fronto-parallel plane in the scene), this aggregation introduces errors into the values stored in the DSI. This is because all the pixels in the two windows do not actually correspond, e.g. due to foreshortening. Thus, any methods (including simple parabola fitting and Shimizu and Okutomi's approach) which utilize a DSI aggregated with simple rectangluar windows to find a sub-pixel minimum will be biased.

Our approach circumvents this problem by locally adapting the shape of the aggregation window W *in the original images* to determine the best sub-pixel disparity value. We only use the DSI as described above, which can be implemented very efficiently when aggregated with simple rectangular windows,

[1] An exception is the parabola fitting example in Figure 1. This result was in fact generated using absolute differences to highlight the pixel-locking effect for illustrative purposes. Note the more pronounced peaks as compared to results using squared differences later in the paper.

in order to get the initial integer disparity estimates, $d_{int}(x, y)$. These estimates provide good starting points for positioning corresponding windows in the left and right image. We then consider the finding of sub-pixel offsets, $d_{off}(x, y)$, as well as better shapes for each $W_{(x,y)}$, to be a classic optical flow or template tracking problem, as explained in the remainder of this section.

The estimation of optical flow, in the standard sense, is the determination of the motion of each pixel in an image over time. In other words, we wish to find a displacement vector field $\big(u(x, y), v(x, y)\big)$ which maps the pixels in an image at one instant in time to their observed locations at the next instant in time:

$$I_t\big(x + u(x, y), y + v(x, y)\big) = I_{t+1}(x, y)$$

The two images in our case are not from a temporal sequence but instead are the simultaneously-captured images from the left and right cameras. Because they are rectified, only horizontal displacement (equivalent to disparity) must be considered, and thus $u \equiv d$ and $v \to 0$. In addition, we already have an initial estimate for the coarse, integer portion of the disparity. Let $x_d = x - d_{int}$ be the corresponding horizontal position in the right image for position x in the left image, according to the initial integer disparity estimate. We are only seeking the sub-pixel update to this initial estimate, yielding the following relation:

$$I_R\big(x_d - d_{off}(x, y), y\big) = I_L(x, y)$$

A first-order approximation yields

$$I_R\big(x_d - d_{off}(x, y), y\big) = I_L(x, y) + d_{off}(x, y)\frac{\partial I_L(x, y)}{\partial x}.$$

In order to solve for $d_{off}(x, y)$, we can again consider a window W around the location (x, y) and solve a least squares problem:

$$d_{off}(x, y) = \arg\min_d \sum_{(i,j)\in W_{(x,y)}} \big(d \cdot I_x + I_e\big)^2$$

where $I_x = \frac{\partial I_L(i,j)}{\partial x}$ and $I_e = \big(I_L(i, j) - I_R(i_d - d, j)\big)$ for notational simplicity. Note that we compute I_x simply using finite central differences.

At this point, we are still computing the *single* disparity offset which will minimize the difference between all pixel intensities within corresponding windows in the left and right images. If we drop the assumption that the disparity is constant within W and instead allow it to be a (locally) linear function of the (i, j) coordinates within the window, we can handle the case that the scene is locally planar and the variation of scene depth within the window is small compared to the distance to the camera. Thus we replace d in the equation above by a linear function of local window coordinates i and j:

$$d(i, j) = ai + bj + c$$

Now our goal is to solve for the plane parameters (a, b, c), again in a least squares framework. After some algebraic manipulation, this can be written as

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = -\begin{bmatrix} \sum i^2 I_x^2 & \sum ij I_x^2 & \sum i I_x^2 \\ \sum ij I_x^2 & \sum j^2 I_x^2 & \sum j I_x^2 \\ \sum i I_x^2 & \sum j I_x^2 & \sum I_x^2 \end{bmatrix}^{-1} \begin{bmatrix} \sum i I_x I_e \\ \sum j I_x I_e \\ \sum I_x I_e \end{bmatrix}$$

where the summations are over all (i, j) in the window W.

Solving the above system for (a, b, c) not only allows us to compute the best sub-pixel offset for the pixel at the center of the window, which corresponds to $d_{off}(x, y)$, but also the spatially-varying updates for *all* points within W. Thus, we effectively have an improved guess for the *shape* of the window over which to aggregate the error I_e for position (x, y). We can now repeat the estimation using the improved correspondence window to better our estimate of d_{off} (and W), iterating until convergence. In cases where the above approach does not converge, we resort to standard parabola fitting. In practice, this is only necessary for approximately 1% of the pixels in the image.

Iteratively re-estimating the offset has the additional advantage of allowing offsets greater than one pixel to be found as the window warps and shifts across the right image in search of the lowest matching error. This has the effect of potentially correcting initial integer disparities that are wrong by more than a single disparity level. In fact, corrections on the order of half the width of the aggregation window are possible. Large initial errors of more than half a pixel are possible due to foreshortening when using simple rectangular windows to compute the DSI. Note that in such cases, the disparity minimum in the flawed DSI – whether found to integer or sub-pixel accuracy – *does not actually correspond to the true disparity in the scene*. Thus parabola fitting, which cannot produce sub-pixel offsets larger than ± 0.5, is doomed to fail from the outset in these cases. Our approach can succeed due to its use of adaptive windows in the original imagery rather than using the already-biased DSI.

Also note that between iterations, only the position and shape of W in the *right* image change. Thus, the only term which must be updated in the linear system above is the error term I_e. The other terms – including all those of the expensive matrix inverse – need only be computed once, saving significant computation. We can therefore consider each pixel and its local window in the left image as a *template* for which we are seeking the best position and window shape in the right image to minimize an aggregated matching cost. The above formulation is therefore exactly that of standard affine Lucas-Kanade (LK) template trackers, but we are treating *every pixel* as a template and restricting tracking to be along scanlines since the images are rectified.

The use of discrete windows for summing matching error requires a few additional practical considerations. First, to avoid artifacts due to sharp changes in matching cost as the window's size or position change, we use a Gaussian-shaped weighting function. In addition, we ignore any pixels (by zero-weighting them) which are marked as occluded or invalid in the initial integer disparity result (e.g. those that fail a left-right consistency check). In our implementation, such pixels are marked by a disparity of -1. Finally, if the window straddles an occlusion boundary, the center pixel's computed offset will be influenced by pixels from a different physical surface in the image. Since we have an initial estimate of disparity (and thus the occlusion boundaries in the scene), we also ignore any pixels in the window whose initial integer disparity differs radically from that of the central pixel whose offset we are

computing. The weights can thus be written as:

$$\omega(i,j) = \begin{cases} 0 & |d_{int}(i,j) - d_{int}(0,0)| > T \\ 0 & d_{int}(i,j) = -1 \\ \exp(-\frac{i^2+j^2}{2\sigma^2}) & otherwise \end{cases}$$

We use a σ equal to half the window width and let T=2 for our results. We also normalize so that the total weight for each window is one.

Strictly speaking, the formulation in this section assumes perfect brightness constancy between corresponding windows in the left and right images. In reality, the captured intensities for corresponding regions in each camera may vary substantially. This may be due, for example, to differing camera or framegrabber gain settings or to the slightly differing viewpoints of the two cameras. A common solution is to preprocess the images with a Laplacian filter, usually implemented as a difference of Gaussian-smoothed images. Unfortunately, this approach will smooth across occlusion boundaries, resulting in problems similar to those discussed above with windows which contain pixels from two different scene surfaces. We instead use a difference of bilaterally-filtered images [8], [9], as this better preserves edges in the images.

III. Related Work

Perhaps the first documentation and analysis of the pixel-locking effect was in [4] (called "linearization error" in that work), which also studied errors from ground plane foreshortening, window effects, and vertical misalignment. They recommend fitting a quartic instead of a parabola to reduce pixel-locking, but the improvements are not dramatic. We chose to compare our work to the more recent method of Shimizu and Okutomi [7] discussed above. Both papers derive analytical functions for the pixel-locking error with very similar shapes, though the formulation in [4] is much more compact.

Also closely related is the excellent analysis in [10]. There, issues surrounding appropriate sub-pixel sampling when creating the DSI are addressed, but an explicit analysis of pixel-locking is not provided. In addition, we seek a method which can be applied as a post-processing "fix" to existing (real-time) integer disparity estimation techniques. Their approach, which is also potentially expensive in memory and computation, modifies the disparity estimation process from its outset. A similar approach to ours, which utilizes local image gradient information, is discussed in [11], but in a different context, with a very different formulation, and without specifically addressing pixel-locking.

Various authors attempt to handle the well-known foreshortening problem on the ground plane by some form of pre-warping of one image of the stereo pair ([1], [2], [3], [4]). This is an efficient approach since simple, rectangular windows may still be used for disparity estimation, but it only works for a single plane in the image (i.e. the ground plane) and may require some knowledge of that plane's orientation (or a search for it). The work of [12] addresses sources of and fixes for stereo errors on *horizontally* slanted surfaces, but does not discuss foreshortening on the ground plane or provide an explicit analysis of pixel-locking. As discussed above, our approach is more general than these methods as it is not restricted to planes of a specific orientation.

There do exist other methods which attempt to estimate sub-pixel disparity from the beginning rather than adjusting initial integer results (e.g. [13], [14], among others). These methods, which fit smooth parametric surface patches to an over-segmented scene, can be quite computationally expensive. For our approach we chose to compute initial integer disparities and then refine them for computational reasons: the cost of adaptive windows at all locations and all disparities is prohibitive. We therefore assume that the initial integer guesses will be close to the true optimal disparity.

Adapting correspondence windows' positions or shapes is a well studied class of techniques and is often used to improve stereo results ([15], [16], [14], [17], [18] among others), but mainly in the context of improving performance near occlusion boundaries. To our knowledge, this is the first work which specifically evaluates their efficacy for combating pixel-locking in sub-pixel disparity estimation.

IV. Results

We compare our approach to simple parabola fitting as well as our implementation of the error compensation approach of Shimizu and Okutomi. To allow quantitative analysis, we first evaluate results on synthetic imagery. The left image of a synthetic stereo pair along with the corresponding ground truth disparity map are shown in Figure 2, depicting a rectangular room with various textures mapped to the planar ceiling, floor, and walls. Because the four surfaces in the room are planar, the true distribution of (sub-pixel) disparities along each should be uniform. (Note that any stair-stepped pattern visible in the disparity map is due to color quantization; the disparities do indeed vary smoothly.) A histogram of ground truth sub-pixel disparities for a region on the room's ceiling is shown at the top of Figure 3.

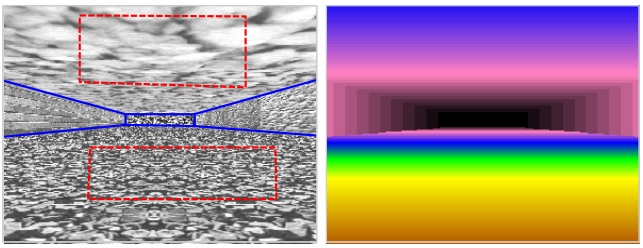

Fig. 2. Left image of a texture-mapped synthetic room and the corresponding ground truth disparity map. (The blue lines are for visual aid only, to help distinguish the surfaces in the room. The red dashed rectangles indicate the ceiling/floor regions used for analysis – see text.)

Initial integer disparities were computed for the pair as described above. Windows for all results provided were 7×7 pixels. A histogram of the resulting integer disparities for the same ceiling region in the synthetic room is shown in Figure 3, with spikes at each corresponding integer disparity. After applying parabola fitting to estimate sub-pixel disparity, we see in the same figure the typical pixel-locking effect as peaks of sub-pixel disparity. Applying the approach of Shimizu and Okutomi improves the "peakiness" somewhat, and using our

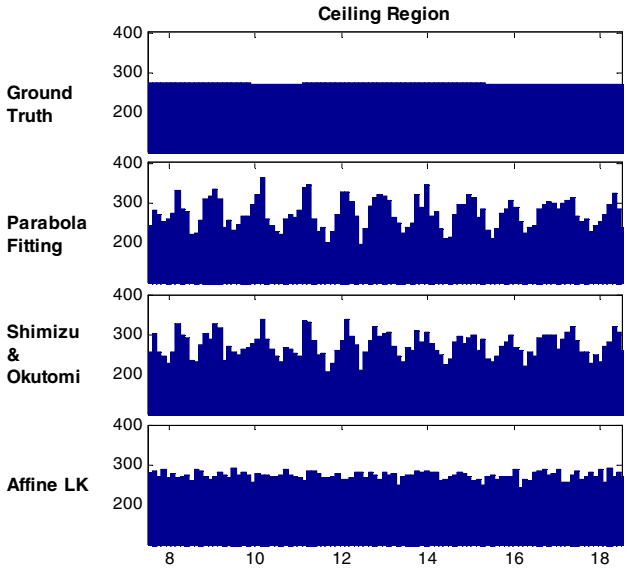

Fig. 3. Histograms of sub-pixel disparity from a region on the ceiling of the synthetic room.

LK approach *with constant windows* yields similar results (not shown). But once we enable affine warping of the windows, we see the more drastic flattening of the histogram shown at the bottom of the figure, which more closely resembles the ground truth distribution.

A similar comparative analysis of the disparity distributions for a region along the floor of the room is provided in Figure 4. Parabola fitting and Shimizu and Okutomi's approach produce very ragged, peaked distributions. Again we see the best attenuation of pixel-locking when using our affine LK approach. Some of the final warped window shapes from a region on the floor are shown in Figure 5.

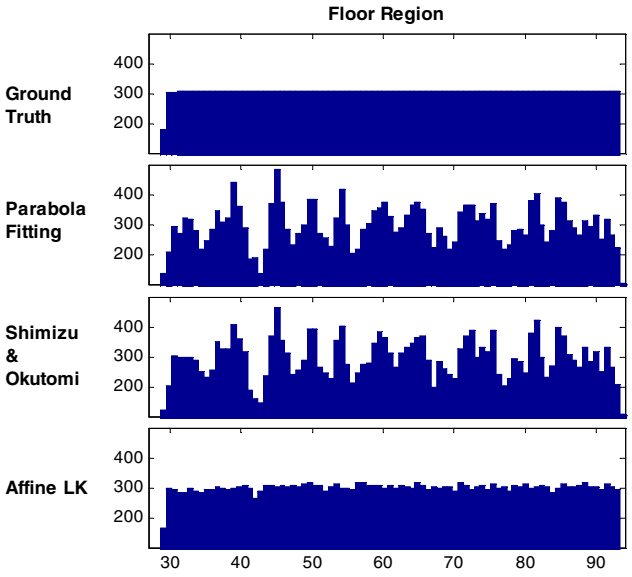

Fig. 4. Histograms of sub-pixel disparity from a region on the floor of the synthetic room.

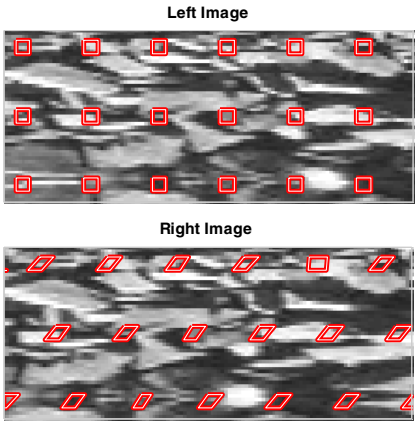

Fig. 5. Corresponding windows with affine adaptation enabled.

In addition to comparing disparity *distributions*, which shed light on the pixel-locking performance, it is important to check the actual disparity *errors* as well. For the initial integer disparity map shown in Figure 6 (a), consider its absolute error versus the ground truth disparity map, shown in (b). The errors have been capped at 0.5 to de-emphasize outliers and highlight initial smaller errors. Note that most of the errors on the floor *start* larger than 0.5. This is most likely due to signficant foreshortening, as the synthetic camera is positioned fairly close to the ground plane (the disparity gradient on the floor is approximately eight times higher on the floor than the ceiling). As discussed above, this high initial error does not bode well for parabola fitting, as we shall see. The error maps after sub-pixel disparity estimation by parabola fitting, the approach of Shimizu and Okutomi, and our affine LK method are shown in (c), (d), and (e), respectively. Note that the errors on the floor are significantly reduced only when using our approach.

In Figure 7, we compare the RMS error over the ceiling and floor regions for the various approaches. To suppress influence of outliers, we ignore pixels whose initial integer disparity error was greater than 3 when computing the following RMS values (note that the choice of the threshold does not radically alter these results). For the ceiling, both parabola fitting and the method of Shimizu and Okutomi do reduce the error from the initial integer estimates by about 65%. But our affine LK approach reduces the error even more: by 78% . On the floor, where the errors are much higher initially, both methods based on parabolas are quite limited in their ability improve the integer estimates. The error is only reduced by about 6% using those approaches. But with the adaptive window capability of our approach, a dramatic 86% reduction in error is possible.

Next we compare results on a real stereo image pair, taken from an outdoor autonomous vehicle. At the top of Figure 8, the left image of the pair is shown alongside the initial integer disparity map. Below these, histograms of estimated sub-pixel disparity are shown for the selected ground region using parabola fitting, Shimizu and Okutomi, and affine LK. We do not have ground truth for this pair, but because the ground is roughly planar, we can expect a smooth distribution of disparity. Once again, the pixel-locking effect is clearly visible for simple parabola fitting. Shimizu and Okutomi's

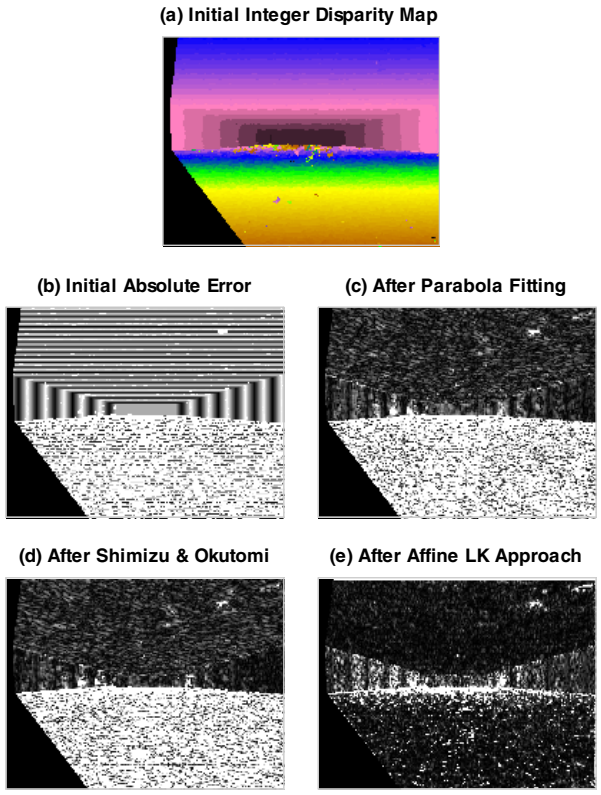

Fig. 6. Absolute disparity errors for the synthetic room imagery, capped at 0.5. The large initial errors on the floor are corrected only using our affine LK approach.

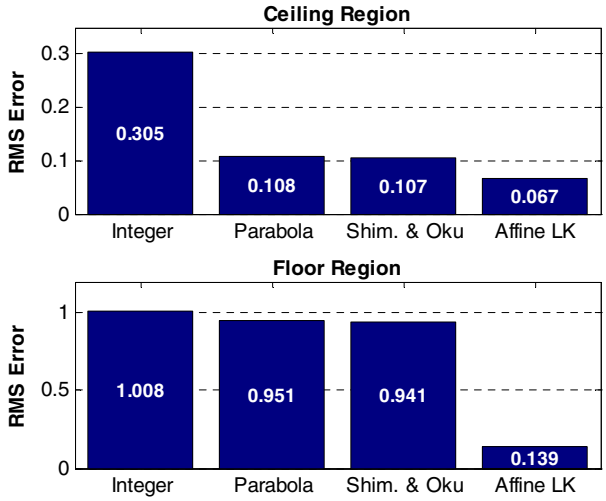

Fig. 7. RMS sub-pixel disparity errors from the floor and ceiling regions of the synthetic room imagery.

method shows marked improvement for this pair, but our affine LK approach still produces the smoothest histogram with the smallest peaks.

An additional example on real data is provided in Figure 9. The error compensation of Shimizu and Okutomi somewhat reduces the peaks of the standard parabola fitting method, but the affine LK method produces a smoother result.

In Figure 10, we compare the artificial rippling of the ground plane in the 3D reconstructions of the data in Figures 8 (left column) and 9 (right column). Each row provides an overhead view of a reconstruction using a different method for computing the sub-pixel disparities. The approximate camera viewing directions are indicated by white arrows in the top row for reference. Our affine LK approach results in the least undesired artificial structure on the ground plane, while leaving only the bumps which are due to actual obstacles or terrain in the scene. In addition to the methods and parameters discussed above, we have also included for comparison the results for parabola fitting when using absolute differences to construct the DSI (top row) and the results for the affine LK approach with larger windows (bottom row). Note in the right column that the larger windows result in a slightly less noisy reconstruction and that details which could be confused with the artifical structure are retained (e.g., the small log in the lower left of the original image data, visible in the lower right of the reconstructions). The incorporation of occlusion information into the window weights helps prevent the larger windows from simply oversmoothing the data, while the affine adaptability prevents errors due to the increased effect of foreshortening with a larger window.

V. Conclusions and Future Work

By iteratively shifting and warping correspondence windows, our proposed affine LK approach for estimating sub-pixel disparity given an initial integer disparity map has shown to be very effective at reducing the commonly-observed pixel-locking effect, especially as compared to standard parabola fitting. In our experience, the performance of the method described in [7] was quite variable, depending largely on image content, while our approach seems to produce consistently less pixel-locking. In addition, our LK method is better able to correct for initial disparity errors which are too large for approaches that rely on parabola fitting. Finally, compared to methods which pre-warp one image to account for foreshortening on a single global ground plane, our approach is more general as it can handle planar structure of any orientation anywhere in the image automatically. This generality may find application in very rough outdoor terrain, where assuming the ground is approximated well by a single large plane could be dangerous, or in man-made (particularly indoor) environments, where much of the above-ground structure is often also planar.

A major thrust for our future work will be the evaluation of computational efficiency of this approach since we are interested in incorporating it into a real-time stereo system. There exists much prior work on efficient (even real-time) implementation of Lucas-Kanade tracking and optical flow (e.g. [19], [20]), so there is reason to believe a real-time stereo vision system using the described approach is possible. Note that the restriction of motion along scanlines in the rectified images reduces computation substantially.

We will also incorporate our approach with the real-time multi-window stereo approach of [18], which will likely improve performance near occlusion boundaries. This would provide better initial integer disparities as well as further information for tayloring the window weights for LK updates.

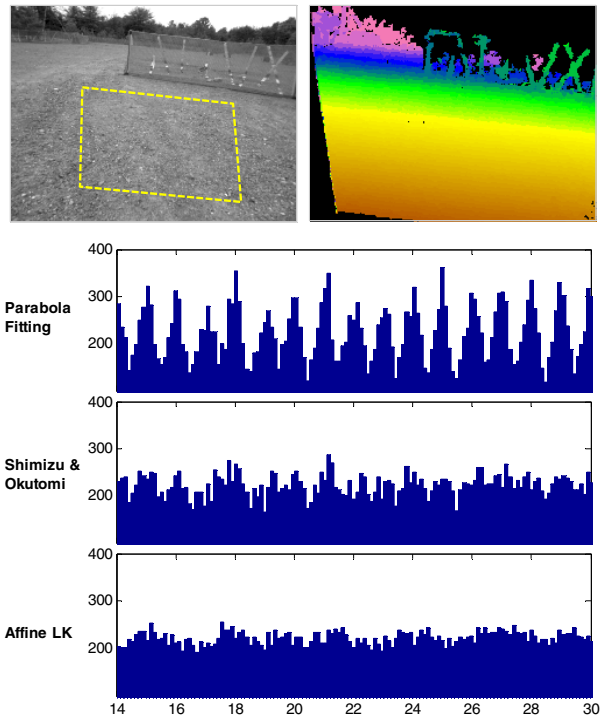

Fig. 8. The left image of a real stereo pair with associated initial integer disparity map and sub-pixel disparity distributions for the ground region (designated by dashed yellow lines). Here, Shimizu and Okutomi's method seems to help, but our affine LK method still produces the least peaked distribution.

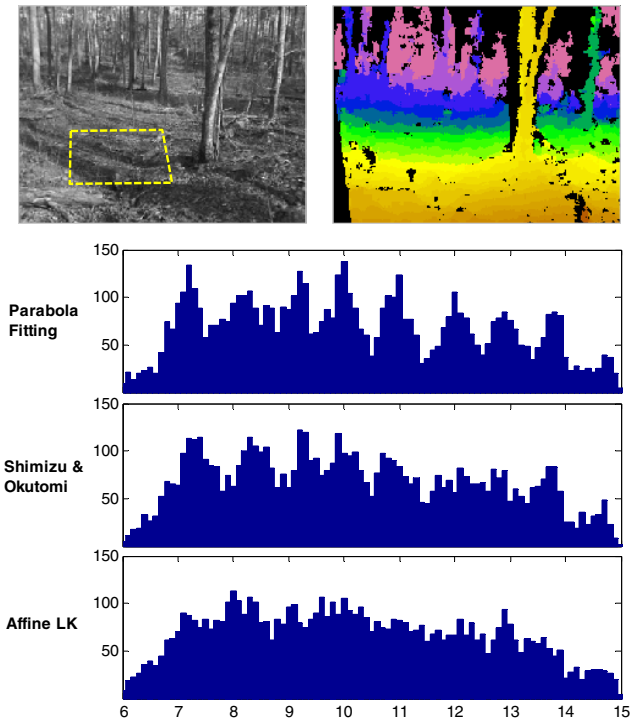

Fig. 9. The left image of a real stereo pair with associated initial integer disparity map and sub-pixel disparity distributions for region on the ground (designated by dashed yellow lines). The method of Shimizu and Okutomi offers minor improvement over parabola fitting. Once again, our affine LK method produces the smoothest distribution.

Finally, we are interested in investigating sensitivity with varying window size and extending the approach to allow for perspective distortions, which could better model the effect of foreshortening by allowing windows to taper. For example, this may reduce the errors on the walls of the synthetic room from Figure 6. This would likely be more computationally expensive, however.

REFERENCES

[1] P. Burt, L. Wixson, and G. Salgian, "Electronically directed "focal" stereo," in *Proc. Int'l Conf. on Computer Vision*, 1995.

[2] T. Williamson, "A high-performance stereo vision system for obstacle detection," Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, September 1998.

[3] R. Mandelbaum, L. McDowell, L. Bogoni, B. Reich, and M. Hansen, "Real-time stereo processing, obstacle detection, and terrain estimation from vehicle-mounted stereo cameras," in *Workshop on Applications of Computer Vision*, 1998.

[4] Y. Xiong and L. Matthies, "Error analysis of a real-time stereo system," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 1997.

[5] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. Int'l Joint Conf. on Artificial Intelligence*, 1981, pp. 674–679.

[6] C. Tomasi and T. Kanade, "Detection and tracking of point features," Carnegie Mellon University, Tech. Rep. CMU-CS-91-132, April 1991.

[7] M. Shimizu and M. Okutomi, "Precise subpixel estimation on area-based matching," *Systems and Computers in Japan*, vol. 33, no. 7, 2002, translated from Denshi Joho Tsushin Gakkai Ronbunshi, Vol. J84-D-II, No. 7, July 2001, pp. 1409-1418.

[8] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. Int'l Conf. on Computer Vision*, 1998, pp. 839–846.

[9] A. Ansar, A. Castano, and L. Matthies, "Enhanced real-time stereo using bilateral filtering," in *Int'l Symposium on 3D data processing, visualization, and transmission*, September 2004.

[10] R. Szeliski and D. Scharstein, "Symmetric sub-pixel stereo matching," in *Proc. European Conf. on Computer Vision*, vol. 2, May 2002, pp. 525–540.

[11] F. Devernay and O. Faugeras, "Computing differential properties of 3-D shapes from stereoscopic images without 3-D models," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 1994, pp. 208–213.

[12] A. S. Ogale and Y. Aloimonos, "Stereo correspondence with slanted surfaces: critical implications of horizontal slant," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2004.

[13] M. H. Lin and C. Tomasi, "Surfaces with occlusions from layered stereo," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2003.

[14] H. Tao, H. S. Sawhney, and R. Kumar, "A global matching framework for stereo computation," in *Proc. Int'l Conf. on Computer Vision*, vol. 1, July 2001, pp. 532–539.

[15] A. Fusiello, V. Roberto, and E. Trucco, "Efficient stereo with multiple windowing," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, June 1997, pp. 858–863.

[16] D. Geiger, B. Ladendorf, and A. Yuille, "Occlusions and binocular stereo," in *Proc. European Conf. on Computer Vision*, 1992, pp. 425–433.

[17] T. Kanade and M. Okutomi, "A stereo matching algorithm with an adaptive window: Theory and experiment," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 16, no. 9, pp. 920–932, September 1994.

[18] H. Hirschmüller, P. R. Innocent, and J. Garibaldi, "Real-time correlation-based stereo vision with reduced border errors," *Int'l Journal of Computer Vision*, vol. 47, no. 1-3, pp. 229–246, April-June 2002.

[19] S. Baker and I. Matthews, "Lucas-kanade 20 years on: A unifying framework," *Int'l Journal of Computer Vision*, vol. 56, no. 3, pp. 221–255, March 2004.

[20] G. D. Hager and P. N. Belhumeur, "Real-time tracking of image regions with changes in geometry and illumination," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 1996, pp. 403–410.

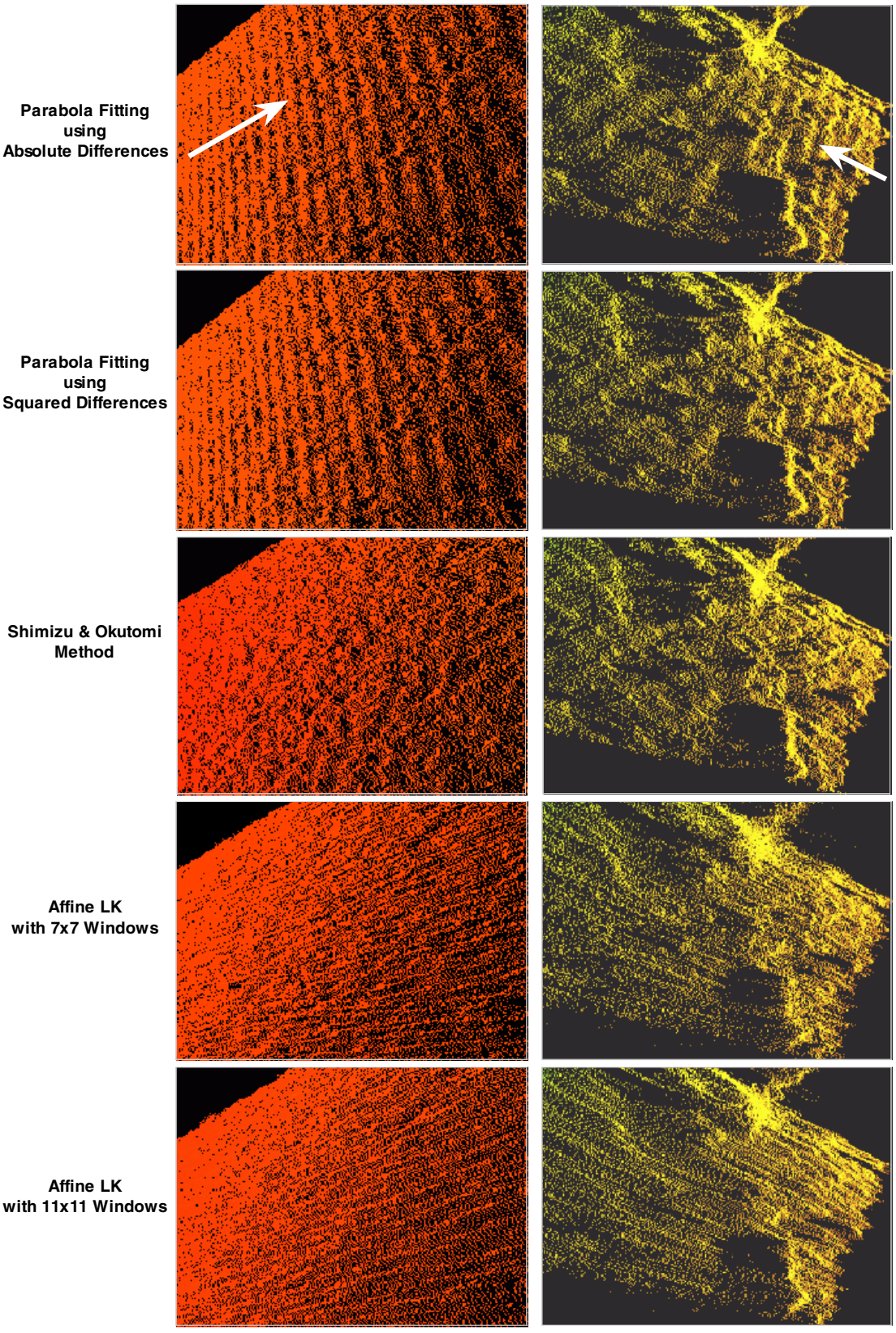

Fig. 10. Overhead views of the 3D ground planes reconstructed from sub-pixel disparity maps for the imagery in Figures 8 and 9, with camera viewing directions indicated by white arrows. The least artificial rippled structure is visible using our affine LK approach. Using larger windows with the affine LK method improves the result even further.