Stereo Matching from Rectified Images

Prateep Mukherjee, Praful Agrawal

May 5, 2015

1 Introduction

Searching for correspondences between a pair of rectified images by matching every pixel on left scanline with every pixel in right scanline is the most naïve way of generating the disparity map. Correspondences are found based on a normalized cross-correlation value computed using neighborhoods of the two pixels being matched. The main drawbacks of this approach are that it is computationally expensive as well as it cannot handle occlusion cases. Moreover, the matching is not very robust in absence of any constraints (such as edge locations or segmented regions) and introduces lot of noise as observed in Assignment 2. In this project, we implemented two techniques to generate disparity map using edge constraints and also a post-processing step to smooth the output disparity map. We compare the results and show that using edge based constraints and post processing help in better disparity estimation. Next section explains implementation details corresponding to the methods used, followed by discussion of results obtained on Middlebury stereo dataset [2][4].

2 Methods and Implementation Details

2.1 Stereo matching using dynamic programming

Ohta and Kanade [3] proposed a dynamic programming based approach to find good correspondences from a rectified pair of images. Their method involve two simultaneous steps of intra-scanline and inter-scanline searching (see Figure 1) based on corresponding edge locations in two images. Both steps involve dynamic programming based approach. Though the algorithm seems promising, it is computationally expensive also acknowledged by the authors in their analysis. During the implementation, we faced some difficulties due to lack of complete details in the paper. We first discuss the challenges in complete implementation of the paper and then how we tackle these challenges to produce fine results:

Challenges:

1. Intra-scanline search involves the computation on cost function for matching edge delimited intervals on the two scanlines where variance of pixel values is used. However, for occlusion cases the cost function depends on a threshold value and authors have missed to comment on this parameter. This parameter being a threshold on variance of gray values can take any value in interval $[0, 255^2]$.

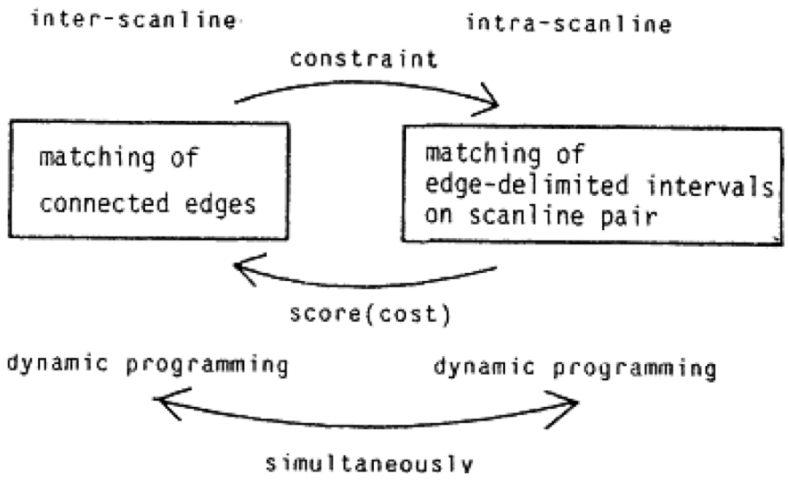


Figure 1: The two steps in stereo matching algorithm proposed by Ohta and Kanade [3].

2. The inter-scanline search heavily rely on the edge linking i.e. finding a connected edge which spans across multiple scanlines within image. An ordering is assigned to all connected edges in the images to find meaningful inter-scanline constraint. However, this ordering can be noisy if an edge crosses on scanline more than once, which is highly probable as edges often have bifurcations. Edge linking strategy was found completely missing from the paper which concludes that an effective inter-scanline search cannot be applied.

Workarounds:

1. For the parameter values, we ran multiple experiments with varying threshold values and fortunately found a correlation among the quality of disparity map obtained and this threshold value, further discussed in results section.

2. In absence of inter-scanline constraints, disparity maps obtained are slightly better than brute force as they involve edge constraints in intra-scanline search. However, the post-processing step being used in this project helps us do away with this step. Authors also discuss that the inter-scanline constraints can be applied as a post processing step and like many existing techniques, we also used prior segmentation maps to apply these constraints in post-processing.

Hence, we implemented dynamic programming based intra-scanline search followed by post processing step to generate disparity map. Other parameters in the algorithm involve threshold on edges to be considered for intra-scanline search. This value depends on the image content and values have been tuned specific to the example images used for experiments.

2.2 Automatic stereo reconstruction of man-made targets[1]

We implement a sequential scan-line based approach for measuring and representing three-dimensional geometry of rectified images. The central problem is that the number of "match-points" required for construction of a single detailed model is very high for high-resolution images. Thus, iterating over every pixel on every scan-line is computationally infeasible. It is therefore desirable to automate the process of correspondence matching. The strategy used in this paper is to match edges in

the stereo pairs, rather than every pixel. The underlying assumption used throughout is that the images consist of planes. The paper reports the demonstration of feasibility of automated stereo measurement over planar surfaces.

Epipolar Geometry: The epipolar geometry in this problem is illustrated in Fig. 2.

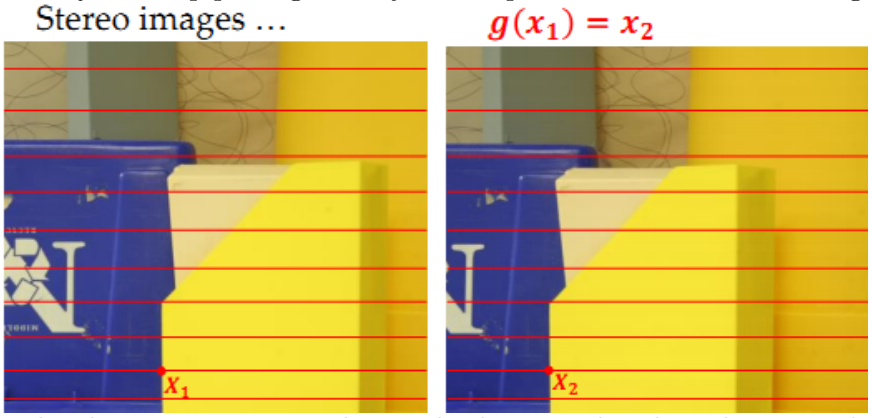


Figure 2: Left and right stereo images with epipolar lines overlayed on them. To show a correspondence, $\mathbf{x}_1$ on left image matches with $\mathbf{x}_2$ on right image. The goal here is to determine $g : \Re \mapsto \Re$ such that $g(\mathbf{x}_1) = \mathbf{x}_2$.

A set of "match lines", or epipolar lines, is overlaid on the two rectified stereo images. A consequence of imaging geometry is that the matching problem can be solved separately for each epipolar line pair. By a process of intersecting three-space rays, matching solutions are transformed into a set of target profiles, such illustrated in Fig. 3.

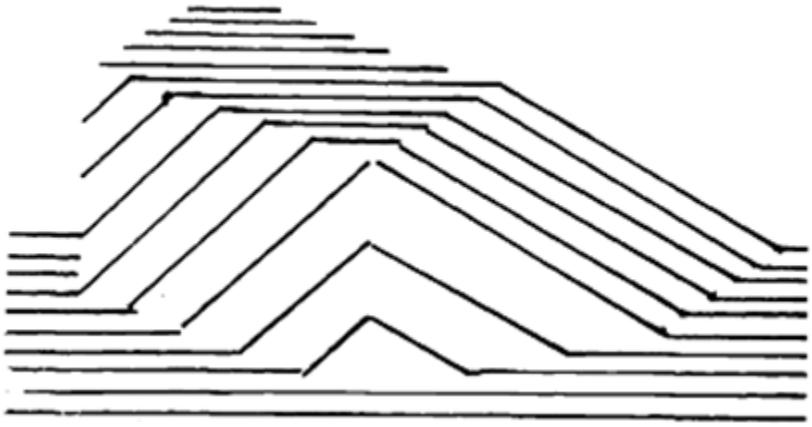


Figure 3: Edge profiles generated for every epipolar line in Fig. 2, scanned from bottom to top.

The matching system, coined as *Broken Segment Matcher* in the paper, is an automated solver for the matching problem for each conjugate pair of epipolar lines. We describe the general idea and the method for *Broken Segment Matcher* in the following section.

Broken Segment Matcher: The broken segment transformation is a graphical representation of matching points along conjugate epipolar lines. The idea is shown in Fig. 4. Let, $\mathbf{X}_1$ be position along one epipolar line in image 1, and $\mathbf{X}_2 = g(\mathbf{X}_1)$ be its matching position in image 2. Notice, that in order to compute the transformation g, $\mathbf{X}_1$ and $\mathbf{X}_2$ need to be coordinates of line-segment

end-points. This results from our assumption of rigid planar geometry in our scenes. The broken segment transformation provides a convenient represetation of the problem features of planar surface sites. **Abrupt changes in slopes** correspond to "breakpoints" while an occlusion corresponds to a horizontal or vertical segment, as shown in Fig. 4.

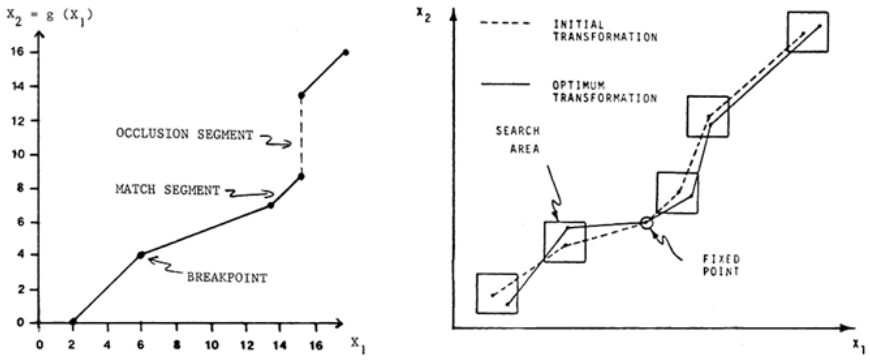


Figure 4: **Left:** Broken Segment Transformation, breakpoints and occlusions are shown. **Right:** Specification of a search with fixed number of breakpoints.

For implementation purpose, we scan the images from bottom to top. As seen from Fig. 3, the positions of the "breakpoints" (a) remain constant, and (b) move to the right, upon transition from one scan-line to another. Therefore, the algorithm proceeds from line-to-line as follows.

1. Find number of "breakpoints" for each line
2. Find optimal transformation for breakpoints in left and right images. This step is similar to using Normalized Cross-correlation(NCC) matching for matching points in Brute-Force method.

Metric: In this section, we discuss the transformation objective used to compute g. Let, λ be the vector of transformation parameters, $f_1(\mathbf{X}_1)$ and $f_2(\mathbf{X}_2)$ be the respective intensities of $\mathbf{X}_1$ and $\mathbf{X}_2$ in left and right images. Therefore, we seek the following optimal transformation.

$$\lambda^* = \underset{\lambda}{\operatorname{argmin}} \sum_{\mathbf{X}_1} C\{f_1(\mathbf{X}_1), f_2(g(\mathbf{X}_1, \lambda)\} + reg(\lambda) \tag{1}$$

where $C(a, b)$ is the NCC score for two intensities. We choose a window-size of 5 pixels in order to compute NCC. It is to be mentioned here that the solution to Eq. 1 is degenerate($\lambda = \infty$) when $\mathbf{X}_1$ corresponds to an occluded section. Therefore, in those cases, a penalty is computed proportional to the occlusion size. The NCC score is chosen in order to compensate for illumination and appearance based differences in the two images.

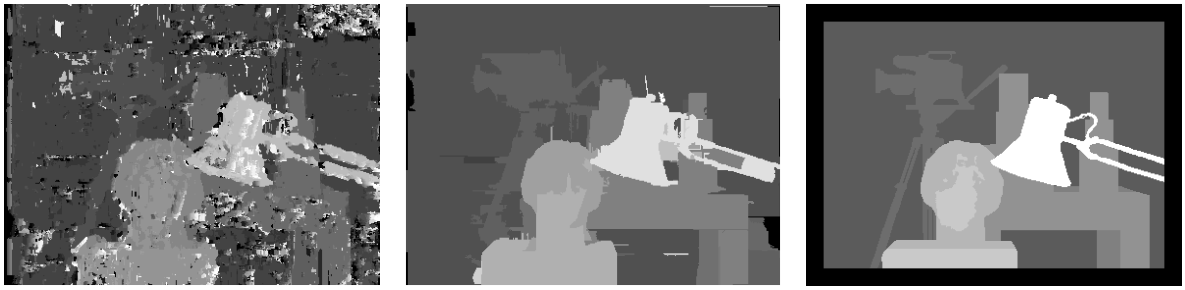
Finally, having obtained the optimal λ, the disparity is computed for every pixel as $D(\mathbf{X}_1) = \mathbf{X}_2 - \mathbf{X}_1$.

2.3 Post processing of disparity map using prior segmentation

It was noted during our experiments, that raw disparity maps weren't quite good due to random noise in the pixels. Therefore, we came up with an idea to smooth the resulting disparity maps using

image intensity cues. The idea is to give similar disparities in a region of pixels which have the same RGB intensity values. In other words, we cluster pixels in the image based on color information, and assign a single cluster, the median of disparity values for all pixels in that cluster. The technique used to perform clustering is called *Mean Shift Segmentation.*The following section briefly describes the method.

Mean Shift Segmentation: The idea here, as mentioned before, is to cluster neighboring pixels in the image with same intensity. For each pixel, mean shift defines a window around it and computes the mean of the pixel. Then we shift the window to the mean and repeat the algorithm till it converges. The above description implies that this is very similar to the classical K-Means algorithm. The only difference here is that we do not know prior the number of clusters needed, and therefore every pixel is assigned a different label initially. Mean shift method is non-parametric iterative algorithm, which is optimized using a kernel-based approach. We use a Gaussian kernel for this project. The Kernel density estimation is done using the classical Parzen window estimate. Fig. 5 shows an example of a raw disparity map, a smoothed version using Mean Shift method and the ground truth generated manually. It is evident that the post-processing technique does a good job here.



(a) (b) (c)

Figure 5: (a) Raw disparity map; (b) Smoothed version; (c) Ground-truth

2.4 Comparison metrics

The disparity maps obtained using above mentioned methods are compared with the ground truth disparity maps available in the dataset. Qualitative as well as quantitative analysis is done in the next section. For quantitative analysis, matching scores like L_1 and L_2 norms can be used, however, the given disparity maps are gray scale images with decimal depth values rescaled to gray scale there would be differences in overall illumination. Therefore, a more robust normalized cross-correlation is computed among the two disparity maps.

3 Results and Analysis

3.1 Qualitative Analysis

- **Brute-force with post-processing:** First, we compare the state-of-art method for stereo analysis. As discussed earlier, post-smoothing of disparity maps with Mean Shift Segmentation improves quality of disparity a lot. Fig. 6 shows two examples of this approach.

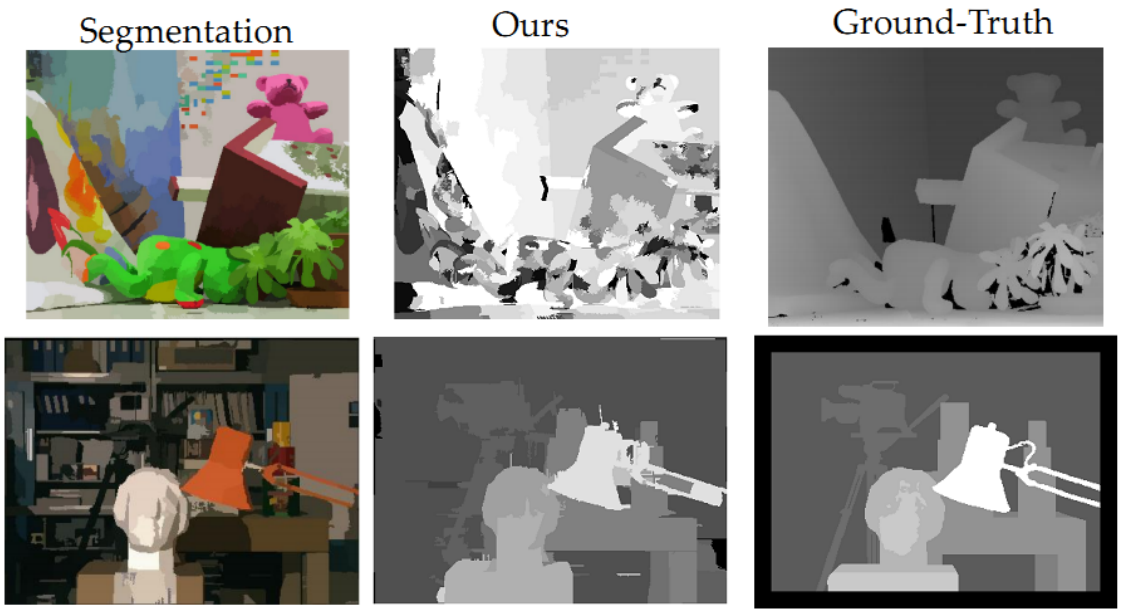


Figure 6: Stereo vision method applied to two datasets, 'teddy-bear' and 'tsukuba'. The figure shows the original left image, results obtained from our method after post-processing and the ground-truth, for both datasets.

- **Ohta and Kanade [3]:** Figure 7 displays the results for different values of threshold in cost function. The result corresponding to the threshold 1000 is better as compared to others as the left part shows correct depth (-ve) as compared to right. For all other thresholds, left depth value seem to be +ve or outwards. Figure 8 compares the ground truth with various images generated during disparity map estimation. It is apparent from the results that without post processing the disparity map is not very informative where as smoothing improved the map significantly. The drawbacks to this approach are that it depends on strong edge estimation and their correspondence, it will not be effective if there are not many edges in the image. For example, if there are no edge locations in a particular scanline pair then the disparity estimation comes down to brute force approach which could give noisy results. Also, if there are too many edges the computation cost further goes up and makes overall process slow.

- **Henderson:**[1] In this section, we compare qualitatively the results obtained using sequential scan-line algorithm described in Section 2.2. Fig. 9 shows the results compared to ground-truth images. The images chosen here intentionally contain plane surfaces(with simply vertical edges). This counts as one of the drawbacks of this method, as the plane surfaces are assumed to be vertical.

  Some **drawbacks** of this approach are as follows.

  (a) Noise-free images.
  (b) Plane surfaces with horizontal and vertical edges only.

3.2 Quantitative Analysis

In this section, we numerically compare our results, with ground-truth images generated manually. One of the parameters used in stereo vision is the window size, inside which we look for corre-

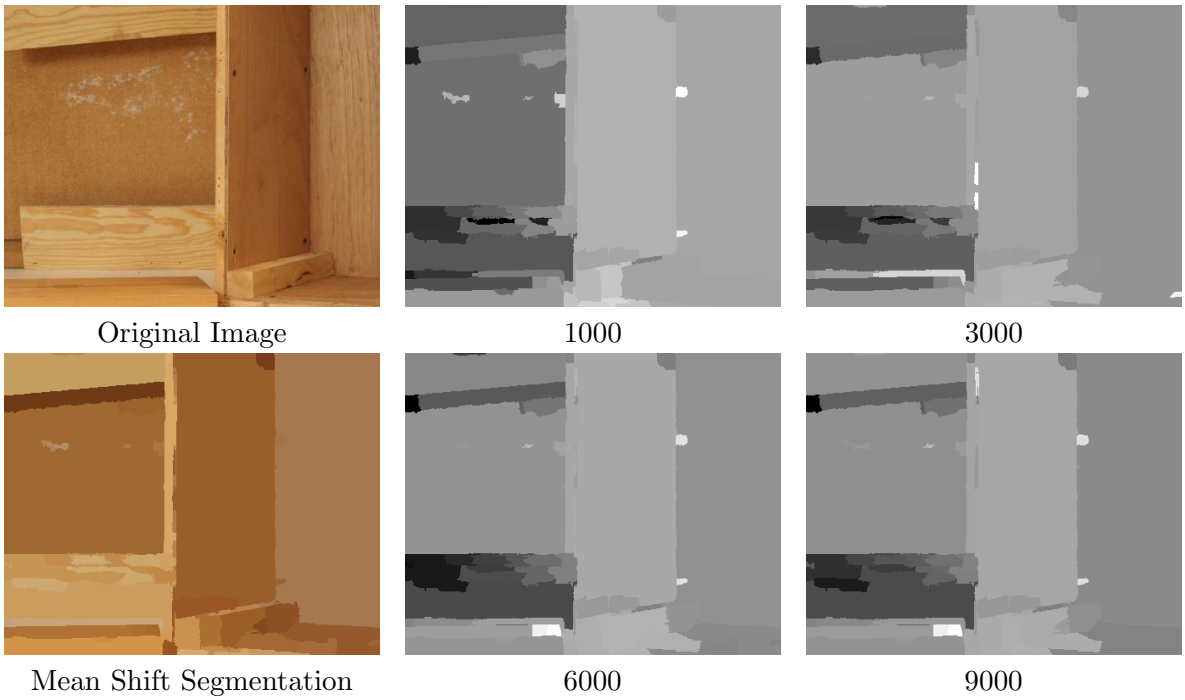


Figure 7: Comparative results for different thresholds in cost function. The disparity maps are corresponding to the left image, mean shift segmentation output is also displayed.

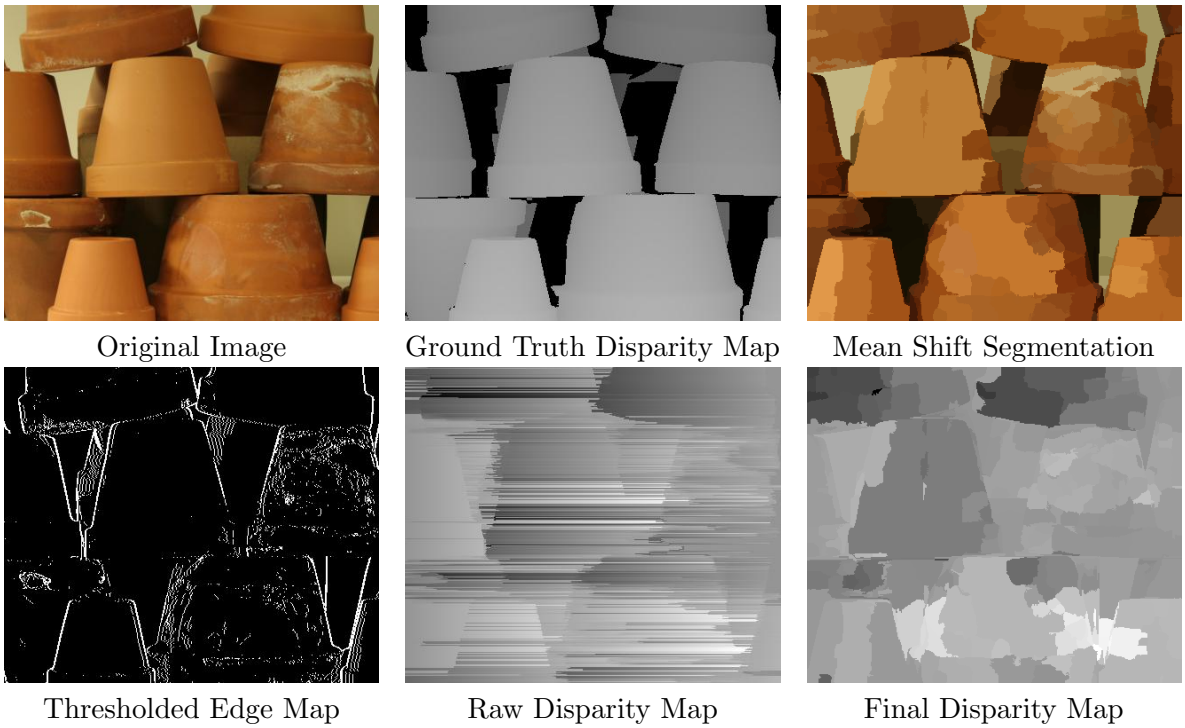


Figure 8: Figure comparing the actual image, ground truth disparity map, mean shift segmentation and outputs at various stages of disparity map generation.

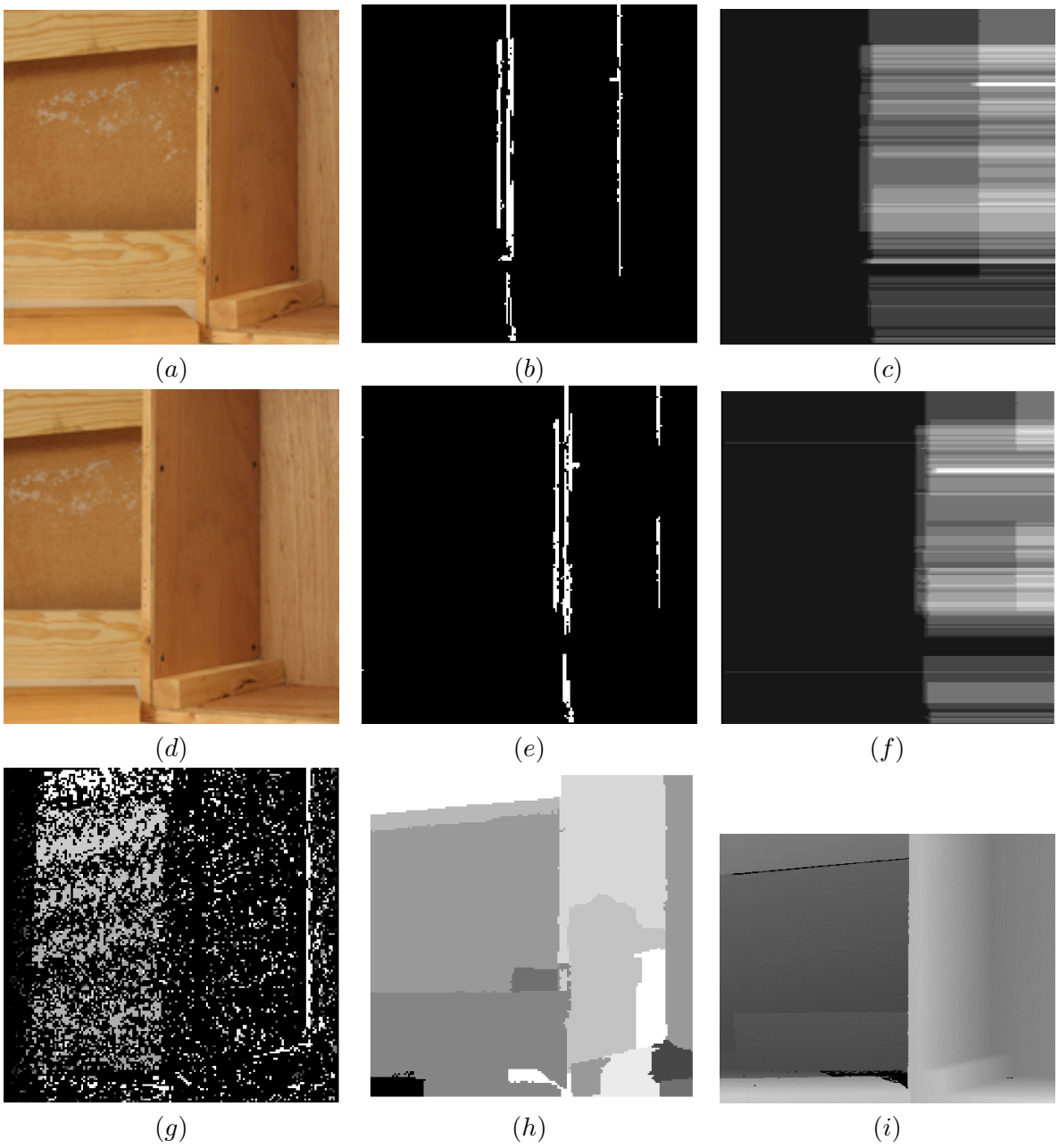


Figure 9: Result from sequential scan-line algorithm. (a,d): Left and Right stereo pairs; (b,e): Vertical edges in both images; (c,f): Edge profiles of each image. (g): Raw disparity map obtained for left image; (h): Disparity map after smoothing; (i) Ground-truth disparity for left image.

spondences. We tested the classic stereo method with post-processing, and compared with the ground-truth data. Fig. 10 shows the difference of our results with that of ground-truth for different window-sizes.

An interesting thing we noted here was that accuracy decreased as we looked at increasing width around every pixel. The Normalized Cross-correlation performed best, as it is robust to different lighting conditions.

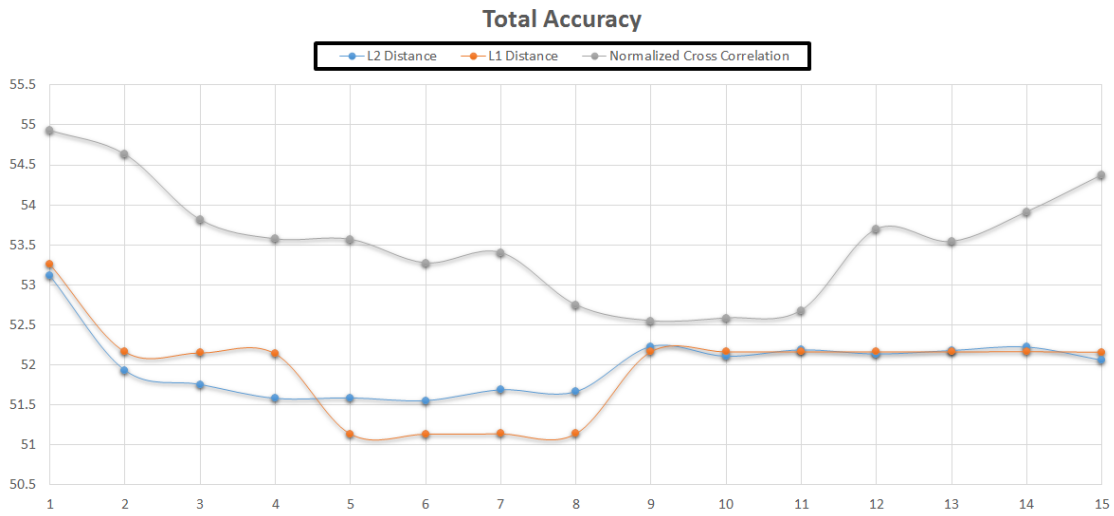


Figure 10: Total accuracy of the brute-force algorithm with smoothing as compared to ground-truths. Different metrics, like L1, L2 and Normalized Cross-correlation.

References

[1] Henderson, R. L., Miller, W. J., and Grosch, C. Automatic stereo reconstruction of man-made targets. In *Huntsville Technical Symposium* (1979), International Society for Optics and Photonics, pp. 240–248.

[2] Hirschmuller, H., and Scharstein, D. Evaluation of cost functions for stereo matching. In *IEEE Conference on Computer Vision and Pattern Recognition* (2007), pp. 1–8.

[3] Ohta, Y., and Kanade, T. Stereo by intra-and inter-scanline search using dynamic programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2 (1985), 139–154.

[4] Scharstein, D., and Pal, C. Learning conditional random fields for stereo. In *IEEE Conference on Computer Vision and Pattern Recognition* (2007), pp. 1–8.