

# Belief Propagation for Stereo Analysis of Night-Vision Sequences

Shushi Guan<sup>1</sup>, Reinhard Klette<sup>1</sup>, and Young W. Woo<sup>2</sup>

<sup>1</sup> The *.enpeda..* Project, The University of Auckland, New Zealand

<sup>2</sup> Dept. of Multimedia Eng., Dong-Eui University, Busan, Korea

**Abstract.** This paper studies different specifications of belief propagation for stereo analysis of seven rectified stereo night-vision sequences (provided by Daimler AG). As shown in [4], Sobel preprocessing of images has obvious impacts on improving disparity calculations. This paper considers other options of preprocessing (Canny and Kovesei-Owens edge operators), and concludes with a recommended setting for belief propagation on those sequences.

**Key words:** performance evaluation, stereo analysis, motion analysis, real-world sequences, driver assistance

## 1 Introduction

Coarse-to fine belief propagation (see, e.g., [6]) is a possible technique for stereo analysis, and it receives good rankings for engineered indoor high-contrast stereo pairs, see [9]. The question arises how this technique behaves on real-world stereo sequences, such as the seven night-vision sequences provided on [2] in Set 1, and described in [8]. See Figure 1 for examples for these sequences. Each sequence contains between 250 and 300 stereo pairs, each image  $680 \times 350$  in 12-bit resolution.



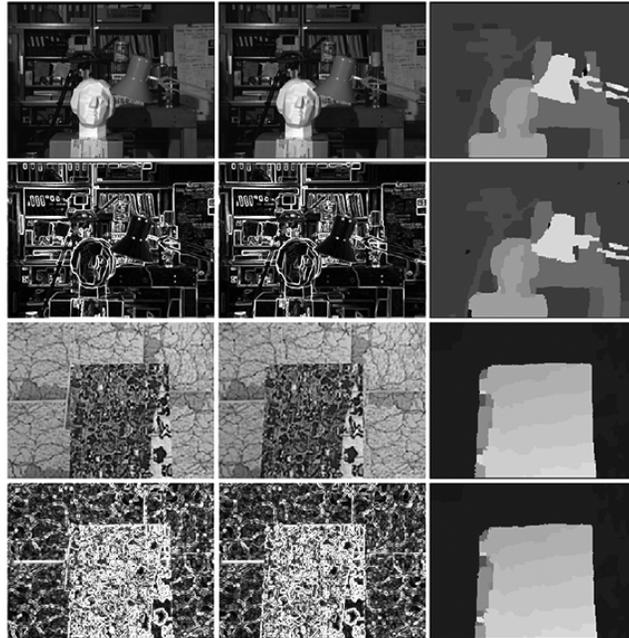
**Fig. 1.** Examples of stereo pairs of Set 1 on the *enpeda..* test sequence website (from Sequences 5, 6, and 7, recorded with 12-bit Bosch night vision cameras).

## 2 Belief Propagation for Stereo Analysis

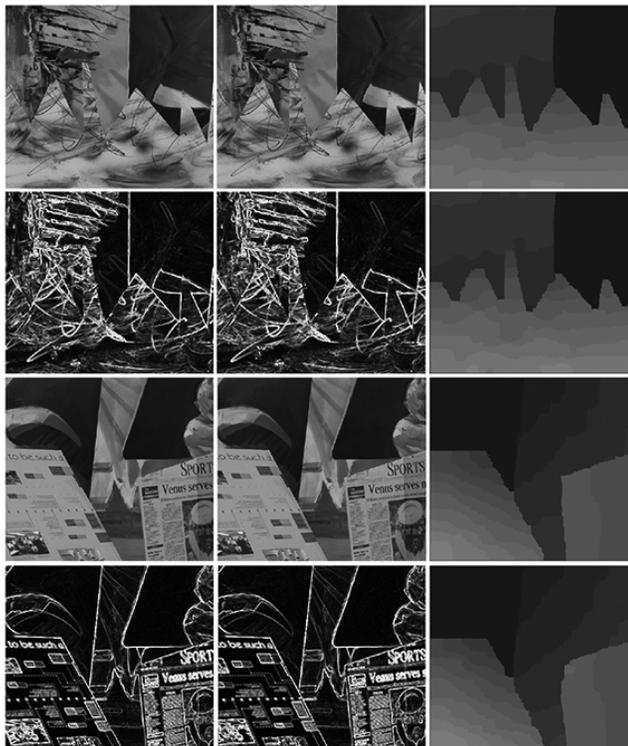
Felzenszwalb and Huttenlocher [3] provide a detailed guide for implementing coarse-to-fine belief propagation (BP); see also [people.cs.uchicago.edu/~pff/bp/](http://people.cs.uchicago.edu/~pff/bp/) for free sources. In our implementation, we decided for max-product, 4-adjacency, truncated quadratic cost function (in difference to [4] where only the simple Potts model was used; the lower envelope calculation may follow [5]), the red-black speed-up method, and coarse-to-fine processing, for more reliable (and also time-efficient) matching. Below we provide further specifications of the recommended BP implementation; those specifications resulted from experimental optimizations with respect to the given seven test stereo sequences. (So far, we have not used any initialization of belief values in message boards by those obtained at time  $t - 1$ , for  $t > 0$ ; initialization is always by the data term using intensity differences at time  $t$ .)

Following [3], we also state two ‘typical’ features of belief propagation. (*Asymmetry*) The strength of message passing from low-difference (homogeneous) areas to high-difference (busy) areas is smaller than the strength of message passing from high-difference areas to low-difference areas. (*Influence of Discontinuities*) Message passing is blocked by discontinuities (such as edges in images).

Due to these observed features, it is not surprising that [4] suggests Sobel preprocessing [12] prior to BP for those real-world sequences. Figure 2 illus-



**Fig. 2.** Upper row: original *Tsukuba* images and BP result. Second row: their Sobel images and BP result. Third row: original *Map* images and BP result. Bottom row: their Sobel images and BP result.



**Fig. 3.** Upper row: original *Sawtooth* images and BP result. Second row: their Sobel images and BP result. Third row: original *Venus* images and BP result. Bottom row: their Sobel images and BP result.

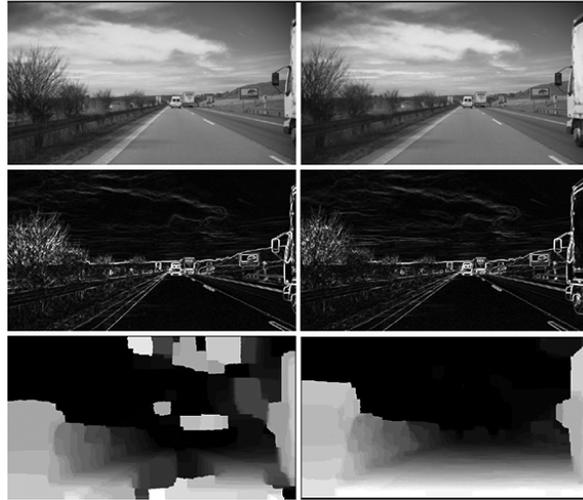
trates the ‘tuned’ BP algorithm on Middlebury stereo examples, without and with Sobel preprocessing. There is actually a slight increase in errors for those engineered images.

As shown in [4], the situation is totally different for the studied real-world sequences; see Figure 4 for an example for Sequence 1. There are (at least) two obvious problems for original images for Sequence 1, namely bad matches due to lack of texture (e.g., middle of the road), and mismatches due to ‘fuzzy depth discontinuities’ (such as in the sky or in trees). Sobel preprocessing contributes towards solutions of both problems.

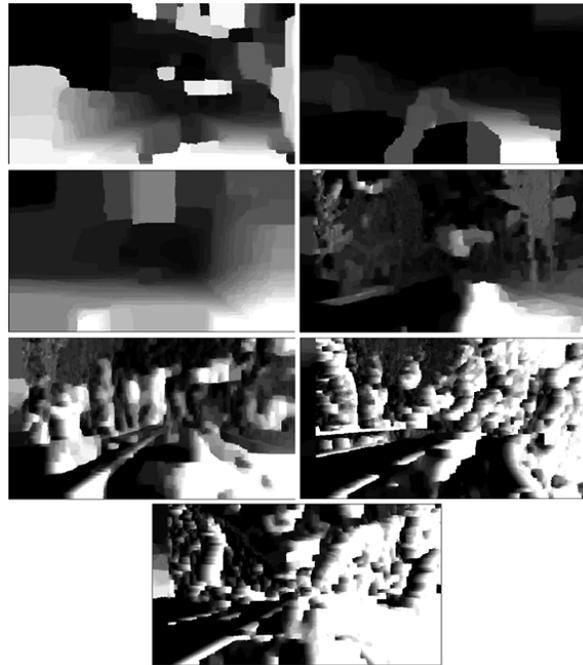
This experience can be made for all the seven sequences. Figure 5 shows samples of depth maps, one for each sequence, without using any edge preprocessing.

**Table 1.** Percentage of bad matches of our BP algorithm on Middlebury stereo pairs, without or with Sobel preprocessing.

Pair	Tsukuba	with S.	Map	with S.	Sawtooth	with S.	Venus	with S.
Error	1.75	1.81	0.31	0.33	0.94	0.95	0.99	1.02



**Fig. 4.** Sample of left and right image for Sequence 1 (upper row), Sobel images of both (middle row), and BP results (lower row) for original images (left) and Sobel images (right).



**Fig. 5.** Samples of depth maps, one for each of the seven sequences (without any edge preprocessing).

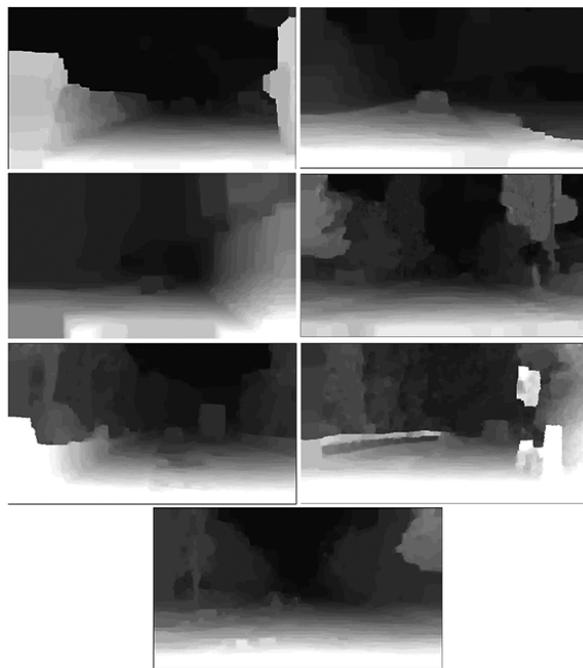
For example, Sequences 5 to 7 show many trees along the road side, and trunks, branches and leaves of the trees cause serious stereo mismatching.

### 3 Comparing Three Edge Detectors

A Sobel edge image is certainly ‘noisy’, but provides this way features or details of original images which allow that the message passing mechanism proceeds more ‘in accordance’ with the actual data. Figure 6 shows typical depth maps obtained for those seven sequences, for the same pairs of frames as already illustrated in Figure 5.

In general, compared to BP on original image pairs, major discontinuities are now, in general, correctly detected. For example, the visual border of a tree may be recovered despite of an obvious fuzziness of image intensities along its visual border. In particular, road and sky are now often accurately located. In most cases, another car is also detected if at a reasonable distance. However, there are still remaining problems. For example, depth maps of first image pairs of Sequence 6 show that the traffic light is not matched correctly (there are ‘two traffic lights’ in the depth maps).

In some images, we cannot identify many depth details, especially in images with lots of trees. Refer to Sequence 6 again for an example. Vertical edges disappear in depth maps. The reason might be that we have chosen a small

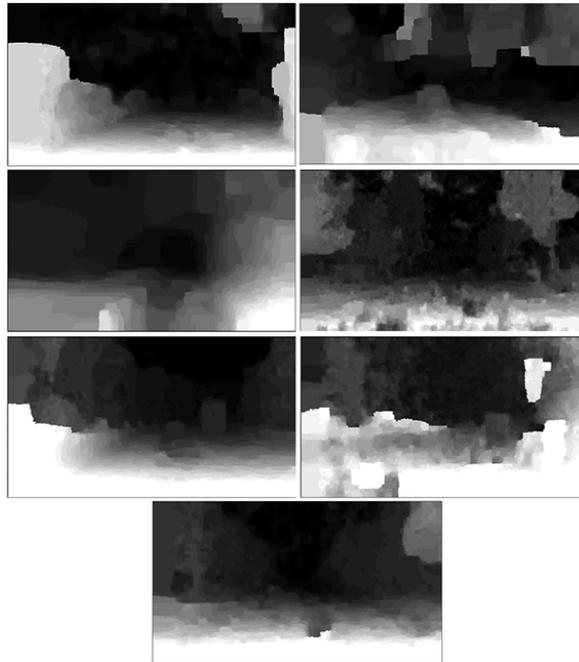


**Fig. 6.** BP results based on Sobel edge images.

discontinuity penalty (see next section) based on an attempt to identify one uniform ‘best’ set of parameters. Using a higher discontinuity penalty in BP produces more edges or details in depth maps, but also more noise or matching errors. Automatic adaptation might be a possibility. In Sequence 5, the road in the generated depth map is not a smooth, even-leveled surface; this is caused by the shadows of the trees on the road which result in horizontal stripes in depth maps. Approximately constant intensity within an epipolar line shows negative impacts, despite the 2D belief propagation mechanism.

We compare effects obtained by using Sobel preprocessing with those possible if using the popular Canny operator [1]. We briefly recall: a Canny edge algorithm blurs the image by using a Gaussian mask; it produces a binary image; it adopts hysteresis between two threshold values to remove edge pixels caused by noise. We use 5 as lower threshold, and 12 as upper threshold. Resulting Canny edge images are certainly not ‘noisy’ (compared to Sobel images). The Canny edge operator ‘aims’ at clear edges in the original images. Figure 7 shows BP stereo results on Canny edge images, for the same stereo pairs as already illustrated in Figure 5.

Obviously, results are worse compared to BP results based on Sobel edge images. For example, depth maps for Sequence 1 show mismatches in the sky (top right), and the road is not a smooth, even surface. Similar problems occur in the depth maps of Sequence 2. However, the truck on the right hand side,



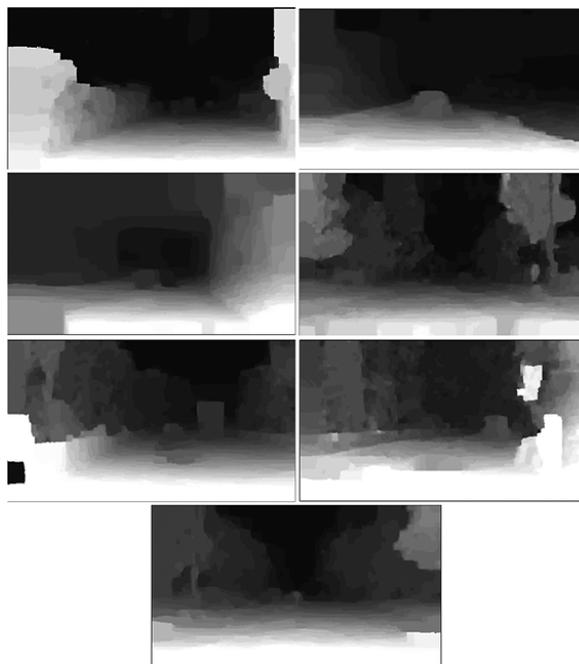
**Fig. 7.** BP results based on Canny edge images.

and the cars in front of the ego-vehicle may still be recovered. Depth maps of Sequences 3, 4 and 7 show that the road is not correctly located. Moreover, cars in these images remain undetected. In Sequence 4, we cannot identify trees along the roadside, the road is often mismatched, and a safety fence in the left lane is not present in the depth map at all. Canny preprocessing was best for depth maps of Sequence 5 (the road is smooth, the safety fence is apparent, and cars at reasonable distances are also detected).

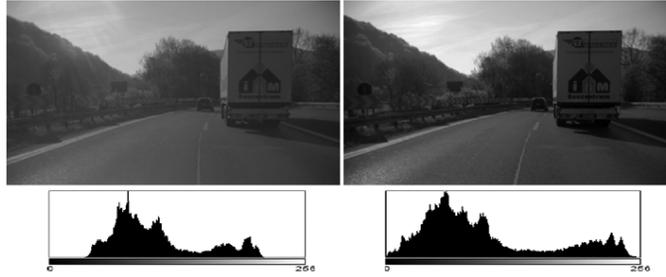
In general we conclude that Canny edge images are filtering out too many details from images. (This was further verified by varying the used thresholds.)

Finally, we selected the Kovese-Owens operator [7, 10, 11] due to its accuracy in general (for image analysis processes depending on detecting actual edge locations). The Canny edge detector often fails in low contrast regions. However, the Kovese-Owens method has a good response for detecting features in low contrast regions. Compared to the Sobel edge image, edges and corners seem to be brighter, thicker, and larger in Kovese-Owens edge images.

The Kovese-Owens edge algorithm is unaffected by changes in image contrast or brightness; it uses phase congruency values, which a range potentially between 0 and 1. We decided to use 0.4 as the phase congruency value. Figure 8 shows BP stereo results on Kovese-Owens edge images, for the same stereo pairs as already illustrated in Figure 5. Compared with BP on original image pairs of the seven sequences, most objects of interest (e.g., cars, safety fences, trees) in the image are now correctly recovered.



**Fig. 8.** BP results based on Kovese-Owens edge images.



**Fig. 9.** An example from Sequence 5: this image pair (with histograms) shows significant brightness differences between left and right image.

However, the Kovesi-Owens depth map results are in general slightly worse compared to Sobel-based results. For example, the road is often not an even surface. The reason might be that the Kovesi-Owens algorithm is here ‘too sensitive’.

Figure 9 illustrates brightness differences between left and right image, which occur from time to time in these sequences. In any of the tested cases, such brightness differences create obvious problems for the tested BP techniques, and Kovesi-Owens seems to adapt best to those cases.

## 4 Tuning of the Algorithm

The selection of ‘best values’ was based on visual (subjective) evaluation and also on selected quantitative performance evaluations. For the latter ones we defined manually rectangular regions in images (such as a face of a truck, or a rectangular approximation of a car driving in front of the ego-vehicle; see Figure 10) and identified their 3D location (by specifying manually disparities of all four corners). This way we also had some approximate ground truth for those sequences available for a more objective performance evaluation.

Figure 11 illustrates the need of large numbers of iterations if no coarse-to-fine technique is applied, and Table 2 shows numbers of iterations for different numbers of levels of the used pyramid, always aiming at about the same performance of Sobel-based BP on the given sequence.

**Table 2.** About equivalent performance of BP with Sobel preprocessing for different pyramid levels (coarse-to-fine strategy) and numbers of iterations.

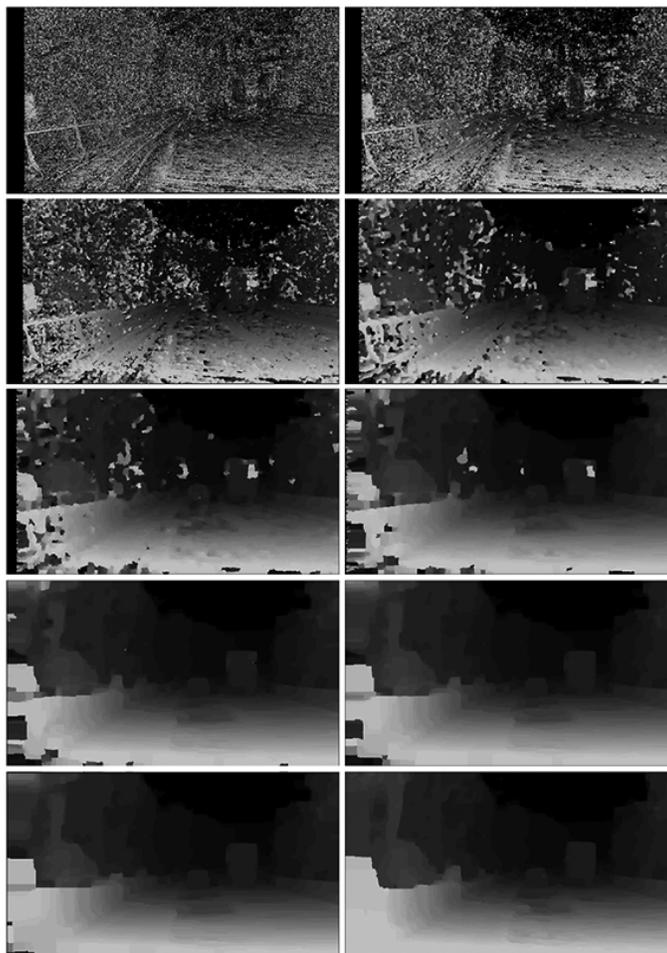
Pyramid levels	Seq. 1	Seq. 2	Seq. 3	Seq. 4	Seq. 5	Seq. 6	Seq. 7
1	410	1210	1120	100	265	150	310
2	100	290	310	35	70	45	100
3	26	120	60	10	25	20	30
4	13	75	20	5	10	10	9
5	5	35	9	3	4	5	5
6	5	15	7	3	3	4	4



**Fig. 10.** Examples of manually specified rectangular regions for approximated ground truth: in original sequences (left) and in Sobel-based BP results (right).

For example, Sequence 2 shows a need for a larger number of levels, to ensure a reasonable reduction of numbers of iterations. Figure 12 illustrates the data in Table 2 by means of function graphs.

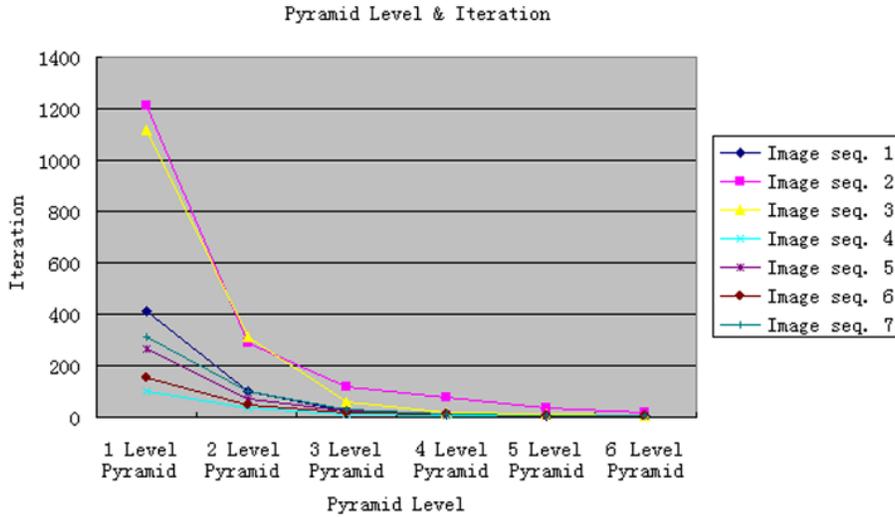
Table 3 shows the actually used numbers of iterations for 5 to 7 levels of the used pyramid (note: using the fine to coarse strategy!) for the seven sequences together with used maximum disparity values (defining the number of used mes-



**Fig. 11.** BP stereo results without using a coarse-to-fine method: BP results (using Sobel preprocessing) after 1 (top left), 2 (top right), 4, 8, 16, 32, 64, 128, 256 (bottom, left), and 300 (bottom right) iterations.

sage boards) and truncation thresholds used for discontinuity and data terms. Running time is the mean per image pair per sequences, including Sobel preprocessing (as our final choice) for each pair of frames (Intel Quad Core 2.4 GHz, 2 Gigabyte memory, NVIDIA Geforce 8800GT video card, WinXP operation system).

Note that constant parameter settings cannot handle variations in the input data, such as different brightness in left and right image. Some type of adaptation needs to be designed.



**Fig. 12.** Performance comparison of BP algorithm with pyramid level and its iteration number for our stereo image pairs.

In general we conclude from our experiments with those seven sequences that a coarse-to-fine strategy not only reduces computation time but also improves accuracy and robustness of stereo results.

## 5 Conclusions

In general, edge preprocessing leads to more accurate stereo correspondence results when using BP. This result, already indicated in [4], was further verified in more extensive research reported in this paper.

**Table 3.** Parameters of the used coarse-to-fine BP algorithm for the given seven sequences (number of message boards, truncation thresholds for discontinuity and for data term), with mean run time for a single stereo pair of one sequence.

Sequence	Max-disparity	Iterations	T (discontinuity)	T (data)	Run time
1	30 <i>pixel</i>	7	11	30	2.9 s
2	35 <i>pixel</i>	7	11	25	3.1 s
3	40 <i>pixel</i>	5	23	20	2.9 s
4	30 <i>pixel</i>	7	20	60	2.9 s
5	30 <i>pixel</i>	5	11	30	2.7 s
6	35 <i>pixel</i>	6	10	30	3.1 s
7	40 <i>pixel</i>	5	11	30	2.9 s

BP stereo results based on Sobel edge images appear to be better in general (for those seven sequences!) than those based on Kovesei-Owens or Canny preprocessing.

At a very general level, we conclude from our studies:

- The edge detector used for preprocessing should reflect and highlight important structural features of the original images.
- It should also not filter any depth-related information or features from the original images.
- It should not add any information or features, by overemphasizing some edge information.

Future work should also cover changes in brightness between left and right image, and some adaptation of parameters in general.

BP has potentials to match real-time requirements in driver assistance systems (DAS), because the BP message updating mechanism at each iteration is actually in parallel, that means multi-CPU hardware is able to reduce run time. Thus, BP is definitely a good candidate for further DAS related studies.

## References

1. Canny, J.: A computational approach to edge detection. *IEEE Trans. Pattern Analysis Machine Intelligence*, **8**:679–714, 1986.
2. *.enpeda.* Image Sequence Analysis Test Site, [www.mi.auckland.ac.nz/](http://www.mi.auckland.ac.nz/) (follow the data link).
3. Felzenszwalb, P. F., Huttenlocher, D.P.: Efficient belief propagation for early vision. *Int. J. Computer Vision*, **70**:41–54, 2006.
4. Guan, S., Klette, R.: Belief-propagation on edge images for stereo analysis of image sequences. In Proc. *Robot Vision*, LNCS 4931, pages 291–302, Springer, Berlin, 2008.
5. Klette, G.: Euclidean distance transform. August 2006, [www.cs.auckland.ac.nz/~rklette/TeachAuckland.html/mm/MI30slides.pdf](http://www.cs.auckland.ac.nz/~rklette/TeachAuckland.html/mm/MI30slides.pdf).
6. Klette, R.: Lecture notes about belief propagation. <http://www.cs.auckland.ac.nz/~rklette/TeachAuckland.html/mm/MI66slides.pdf>.
7. Kovesei, P.: Phase congruency detects corners and edges. *The Australian Pattern Recognition Society Conference: DICTA 2003*, 309–318, 2003.
8. Liu, Z., Klette, R.: Performance evaluation of stereo and motion analysis on rectified image sequences. Technical report, Computer Science Department, The University of Auckland (2007)
9. Middlebury Stereo Website. [vision.middlebury.edu/stereo/](http://vision.middlebury.edu/stereo/).
10. Morrone, M.C., Owens, R.A.: Feature detection from local energy. *Pattern Recognition Letters*, **6**:303–313, 1987.
11. Robbins, B., Owens, R.A.: 2D feature detection via local energy. *Image and Vision Computing*, **15**:353–368, 1997.
12. Sobel, I., Feldman, G.: A  $3 \times 3$  isotropic gradient operator for image processing. Presented at a talk at the Stanford Artificial Project in 1968, unpublished.