A Study on Stereo and Motion Data Accuracy for a Moving Platform

Sandino Morales[1], Young Woon Woo[2], Reinhard Klette[1] and Tobi Vaudrey[1]

[1] University of Auckland, Auckland, New Zealand,
pmor085@aucklanduni.ac.nz, WWW home page: http://www.mi.auckland.ac.nz/
[2] Dong-Eui University, Department of Multimedia Engineering, Busan, Korea

Abstract. Stereo and motion analysis are potential techniques for providing information for control or assistance systems in various robotics or driver assistance applications. This paper evaluates the performance of several stereo and motion algorithms over a long synthetic sequence (100 stereo pairs). Such an evaluation of low-level computer vision algorithms is necessary, as moving platforms are being used for image analysis in a wide area of applications. In this paper algorithms are evaluated with respect to robustness by modifying the test sequence with various types of realistic noise. The novelty of this paper is comparing top performing algorithms on a long sequence of images, taken from a moving platform.

1 Introduction

The main task of computer vision is to use image data recorded by one or multiple cameras to understand the given 3D environment. In particular, stereo algorithms obtain 3D information about the scene geometry, and motion algorithms gather information about the 2D motion of the images. Both types of information are needed to reconstruct the 3D motion of the scene. These algorithms still represent a challenging task for the vision community. For mobile devices the challenge becomes even more difficult; moving background, change in lighting conditions, possible misalignments of cameras, and so forth, make the task of the algorithms even harder. However, the use of cameras has become a popular data sensor for moving vehicles. Vision-based stereo has already been used in a wide variety of vehicles, including wheelchairs (e.g., for the detection of obstacles, unevenness of the ground, detection of stairs and ropes or beams in the air [14]), or for forklifts, where the operator gets valuable information to deal with heavy loads at great heights [15], and in standard cars to assist a driver while driving on a road.

Thus, it is necessary to evaluate the performance of these algorithms to detect which one performs the best in different situations and to encourage their theoretical improvement. Several authors have evaluated stereo algorithms; for example [1] presented one of the earliest evaluations of stereo algorithms and [3] presented one of the most recent and representative evaluation papers so far. Several stereo algorithms were tested, but in both cases the experiments were done with small sets of images. Motion algorithms have also been evaluated, the approach presented in [18] influenced the evaluation of motion algorithms until recently. Now, [17] is the main approach for testing and comparing algorithms online. However, both publications focus on very short image sequences or single stereo frames; none of them analized long stereo sequences.

In this paper we evaluate the performance of stereo and motion algorithms over a long (i.e., 100 stereo pairs) sequence. The analysis of long sequences allows the usage of temporal information (e.g., [6]). In order to test the robustness of the chosen algorithms, we added different kinds of noise to the sequence so that algorithms can be tested under different conditions [13]. As we are interested in testing the algorithms on a mobile platform (a wheelchair or car), we used a long sequence that is publicly available [5], with ground truth for motion and stereo, in Set 2 on the *.enpeda..* Image Sequence Analysis Test Site [4]. This sequence simulates a driving situation.

This paper follows basically [13] with respect to stereo algorithm evaluation, and extends these studies by including evaluations of optic flow algorithms on the same synthetic sequence.

2 Stereo and Motion Algorithms

In this section we briefly introduce the stereo and motion algorithms that we use for our evaluation in this paper.

2.1 Stereo Algorithms

Stereo vision is the process of understanding the $3D$ information of the environment from the available $2D$ data (e.g., a set of 2 or more images), by matching corresponding projections of a $3D$ point in (at last) two images. The algorithms chosen for our analysis are as follows:

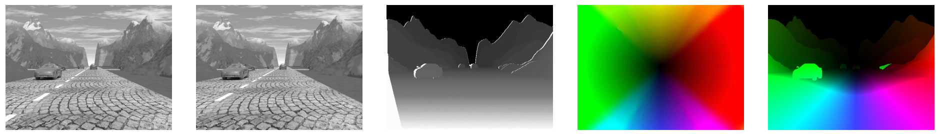
Dynamic programming stereo; we compare a standard algorithm [7] (DP), against one with temporal (DPt), spatial (DPs), or temporal and spatial (DPts) propagation; see [8] for propagation details.

Belief propagation stereo (BP); we use a coarse-to-fine algorithm [9] with quadratic cost function, as reported in [10].

Semi-global matching (SGM) characterizes one of the top performing stereo strategies, see [2]. We chose two cost functions to contrast and compare effects of noise, mutual information (SGM MI) or Birchfield-Tomasi (SGM BT) [11] .

2.2 Motion Algorithms

Motion analysis is estimated from a pair of images taken sequentially. Optic flow algorithm aims to detect the visible displacement of pixels in the image plane to understand the motion of the 2D projection of 3D motion for the visible



(a) Left image (b) Right image (c) GT Disparity (d) Color Key (e) GT Flow

Fig. 1. Stereo image pair #40 of the sequence; (a) and (b) are original left and right images. (c) ground truth data in gray-scale encoding: light = close, dark = far, white = occlusion. (d) color key for encoding optic flow. (e) ground truth optic flow.

objects (and background). The following algorithms are used in our performance evaluation:

Horn-Schunck algorithm (HS) we use the program as available in the OpenCV library [12].

Combination of Local and Global (CLG) optimization [19]. We used an implementation from the *.enpeda..* group (see acknowledgment).

BBPW; is named after the initials of surnames of all the four co-authors of [16]. We used an implementation also from the *.enpeda..* group (see acknowledgment).

3 Evaluation Approach

The algorithms were tested using the original sequence and with the same sequence corrupted with different types and magnitudes of noise. For the stereo algorithms we analyzed the stereo pair at each time frame, and for the motion algorithms only the left images.

In the following we introduce the used data set, the noise that we add to the sequence, how we add this noise, and the quality metrics to evaluate the algorithms.

3.1 Data Set and Visualization

For our experiments we use a long sequence of 100 synthetic stereo image pairs and ground truth data (for stereo and motion algorithms), which are all available on [4], see also [5]. To visualize the stereo results we use gray scale encoding: light for closer objects and dark for objects further away. The color key that we use to visualize the motion results uses hue for direction and intensity for vector size; dark to light means small to large as seen in Figure 1(d).

3.2 Noise

A mobile platform has to deal with non-controlled environments. Thus, we consider it necessary to test the robustness of the algorithms in different situations.

(a) Blurred image. (b) Bright altered image. (c) Gaussian noise added image.

Fig. 2. Corrupted left image #40 (see Fig. 1(a)).

Therefore, we corrupt our data set with three different kinds of noise: brightness differences, blurring, and Gaussian white noise. As a consequence of the movement of the platform, brightness on images can change from one frame to another or even between the left and right image in the same frame of a stereo sequence. Blurring may be caused by differences in the focus of the lenses due to movements of the platform. Gaussian noise is present in images taken, even with modern camera technology. Note that we are aware that this may not be an extensive noise list, but it is sufficient to show the importance of testing algorithms in different conditions.

To alter the brightness of the images, we add a constant brightness c to each pixel of every image. Blurring was applied to the sequence by convolving the images with a Gaussian smoothing kernel of size k. Finally, the Gaussian noise was generated by adding at each pixel random Gaussian (normal distribution) white-noise $\mathcal{N}(\mu, \sigma)$, with a mean μ of zero, and a varying standard deviation σ. The parameters are varied over the sequence and presented in Table 1.

	Left Image		Right Image	
Noise Method	$1 \leq t \leq 50$	$51 \leq t \leq 100$	$1 \leq t \leq 50$	$51 \leq t \leq 100$
Brightness	$c = t - 50$		$c = 50 - t$	
Gaussian Noise	$\sigma = t$	No noise	$\sigma = t$	$\sigma = 101 - t$
Gaussian Blur	$k = 2t - 1$	No noise	$k = 2t - 1$	$k = 203 - 2t$

Table 1. Noise added to image sequence.

To evaluate the motion algorithms we modify the left images using the parameters defined for the right images (Table 1), with the exception of the brightness constant: where $c = t - 52$ is used for even t and $c = 51 - t$ for odd t.

3.3 Quality Metrics

In this subsection we introduce the metrics that we used to evaluate the chosen algorithms. These are commonly used metrics. For stereo algorithms, following [13] and [3], we use the following metrics:

RMS (root mean squared): This is the difference in computed disparity $d(\mathbf{x}, t)$, from one of the algorithms, and the ground truth disparity $d^*(\mathbf{x}, t)$. RMS is defined as

$$R(t) = \sqrt{\frac{1}{N} \sum_{\Omega} \left(d(\mathbf{x}, t) - d^*(\mathbf{x}, t) \right)^2} \tag{1}$$

where N is the number of pixels in the image domain Ω.

% Bad Pixels: This is the number of badly estimated disparities in the image domain, defined as

$$B(t) = \frac{1}{N} \left(\sum_{\Omega} \left(\left| d(\mathbf{x}, t) - d^*(\mathbf{x}, t) \right| > \delta_d \right) \right) \times 100\% \tag{2}$$

where δ_d is a threshold for the allowed disparity error. We use thresholds $\delta_d = 1$ or $= 2$ to determine robustness.

For motion algorithms we use the following two metrics, also used (for example) on [17].

AAE (Average Angular Error): This is the average angle between the ground truth vector $\overline{u}(x, t)$ and the vector $\overline{u^*}(x, t)$ obtained by the algorithm of each pixel of the frame t:

$$A(t) = \frac{1}{N} \sum_{\Omega} \arccos \left(\frac{\overline{u} \cdot \overline{u^*}}{||\overline{u}|| ||\overline{u^*}||} \right), \tag{3}$$

where $|| \cdot ||$ denotes the Euclidean norm.

EPE (End Point Error): This is defined as the average length of the differences of the ground truth vector and the calculated vector of every pixel of the images for each frame:

$$E(t) = \frac{1}{N} \sum_{\Omega} ||\overline{u} - \overline{u^*}||. \tag{4}$$

The metrics used here allow analysis performance similar to the approach used by Middlebury [2], but as we are working with long sequences, we can make statistical inference from the obtained data; such as the mean, zero-mean variance, maximum and minimum, for each error metric over the sequence.

4 Results

In this section we present the results for the stereo and motion algorithms. For the stereo algorithms we present only a few graphs and images, as the detailed results can be found in [13].

(a) Results for AAE over the original sequence.

Algorithm	Mean	St. Dev.	Min.	Max.
BBPW	25.01	26.01	13.87	47.11
CLG	64.42	64.45	62.11	70.28
HS	69.76	69.78	66.74	75.86

(b) Results for EPE over the original sequence.

Algorithm	Mean	St. Dev.	Min.	Max.
BBPW	2.58	2.64	1.69	3.99
CLG	3.09	3.10	2.95	4.18
HS	4.55	4.56	4.41	5.53

Table 2. Results for the original sequence.

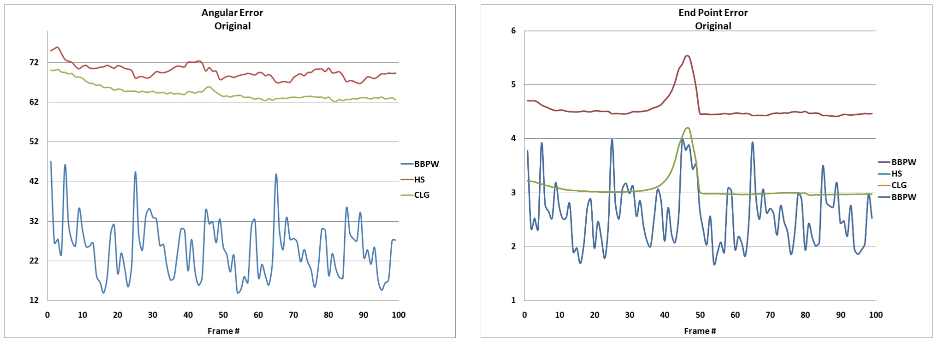
Noise-Free Results: The results obtained with the original sequence are the base for the robustness analysis. For motion algorithms BBPW perform the best with both metrics followed by CLG, see Table 2. In the AAE graph, see Figure 3(a), a considerable difference in magnitude, between BBPW and the other two, is obvious. It is worth to say that for EPE, CLG performs better than BBPW, and its range is the minimum one of all three techniques, see Figure 3(b) and Table 2(b). For examples on images obtained with the three algorithms see Figure 5. For stereo algorithms, the best one for both metrics are the SGM algorithms, followed by BP and finally the DP algorithms, see Figure 6. The difference in magnitudes between the SGM algorithms and the other ones is noticeable. The best algorithm was SGM BT and the best among the DP algorithms is DP closely followed by the other three. For these sequences RMS and BPP show almost the same information. For resultant images of the three top algorithms, with respect to RMS, see Figure 6.

Gauss Blur Results: BBPW outperforms the other motion algorithms with both metrics. With AAE all the algorithms improve their performance with respect to the noise-free results, except for CLG when the blurring is maximum; the change in magnitude is highly noticeable. Compare Figures 3(a) and 7(a) and Tables 2(b) and 3(b). The three algorithms behave the best when the amount of blur is medium and peak when the blur is minimum and maximum. The improvement is most likely because all the algorithms improve their performance on the road area.

For EPE, the behavior of the three algorithms is similar. The maximum error (around frame 50) is where it is expected: when the blurring is maximum. The improvement with a medium amount of blurring is also notorious with this metric for BBPW and CLG, but for HS it is not. See Figure 7(b) and Table 3(b).

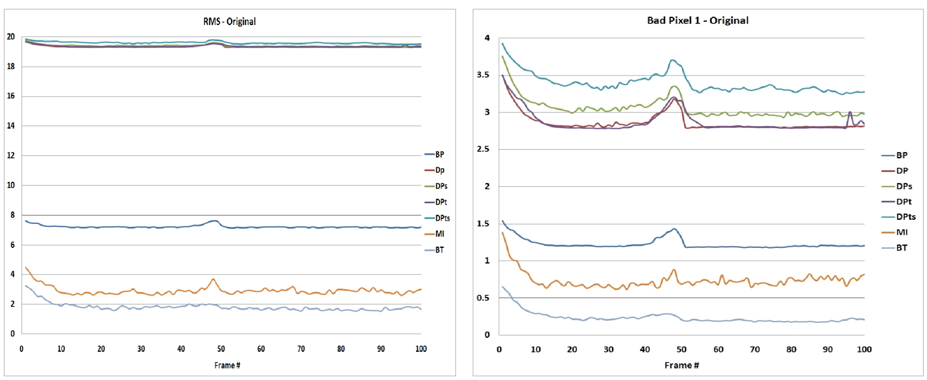
The interesting observation for the stereo algorithms is that when the blurring is in both images, the results are not so bad, but when the blurring is removed from the left image, the results get worse for all algorithms. Again SGM BT is the best, SGM MI seems to have the same problem if both images are blurred, or just one. For BPP the interesting point is that both DP and DPt ranked higher than SGM MI (see Figure 8).

For examples of the resultant images over the blurred sequences with both kind of algorithms see Figures 9 and 10.



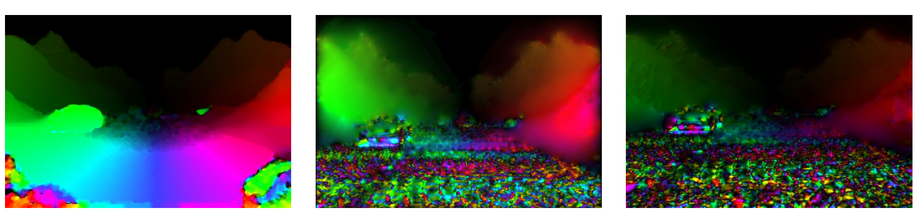
(a) AAE on original sequence.
(b) EPE on original sequence.

Fig. 3. Results of motion algorithms over original sequence.



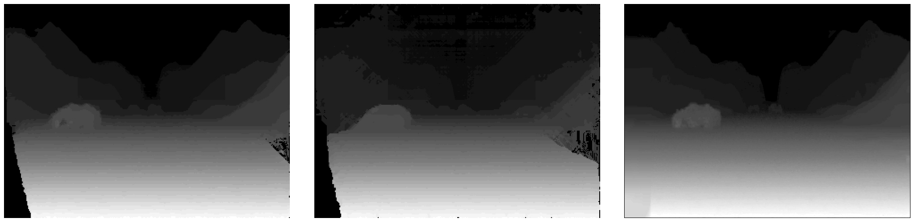
(a) RMS on original sequence.
(b) BPP with $\delta = 1$ on original sequence.

Fig. 4. Stereo results for the original sequence.



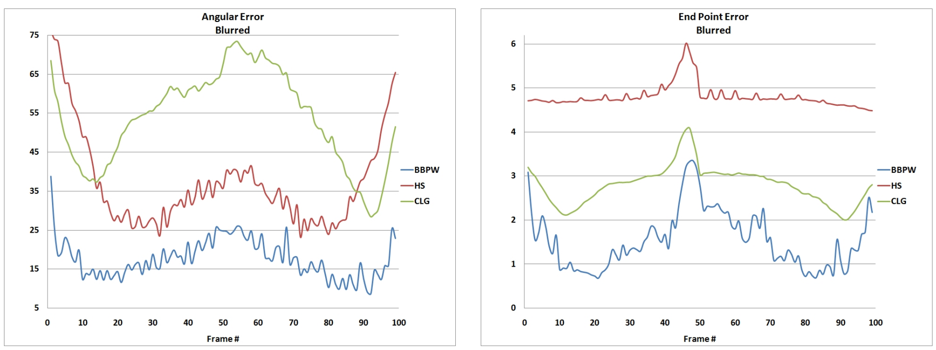
(a) BBPW
(b) CLG
(c) HS

Fig. 5. Results of the analysis with the motion algorithms of frames 40-41 of the original sequence.



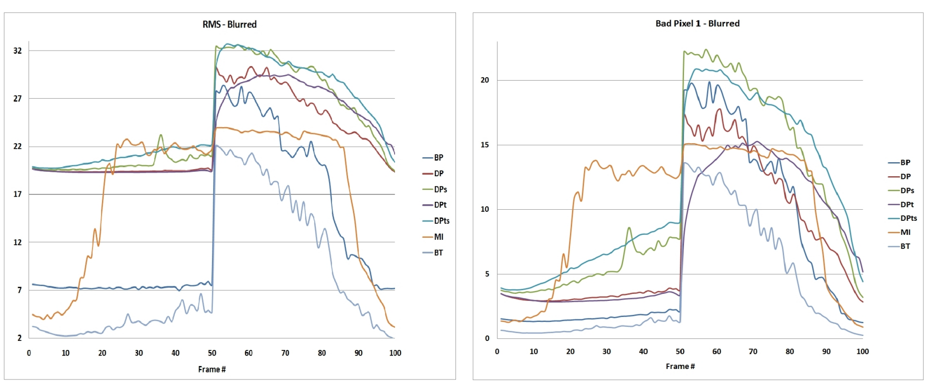
(a) SGM-BT (b) SGM-MI (c) BP

Fig. 6. Results of the three top stereo algorithm (from left to right) for image pair No. 40 of the original sequence.



(a) AAE on blurred sequence. (b) EPE on blurred sequence.

Fig. 7. Results of motion algorithms over the blurred sequence.



(a) RMS on blurred sequence. (b) BPP with $\delta = 1$ on blurred sequence.

Fig. 8. Stereo results for the blurred sequence.

(a) Results for AAE over the blurred sequence.

Algorithm	Mean	St. Dev.	Min.	Max.
BBPW	17.56	18.26	8.65	38.74
HS	37.14	69.78	23.16	76.20
CLG	53.59	54.95	28.45	73.45

(b) Results for EPE over the blurred sequence.

Algorithm	Mean	St. Dev.	Min.	Max.
BBPW	1.53	1.67	0.68	3.34
CLG	2.80	2.83	2.01	4.09
HS	3.29	3.34	2.49	4.72

Table 3. Results for the blurred sequence.

(a) Results for AAE over the bright altered sequence.

Algorithm	Mean	St. Dev.	Min.	Max.
CLG	112.82	114.96	63.57	147.59
BBPW	132.92	140.94	24.89	207.73
HS	142.00	143.87	68.53	167.36

(b) Results for EPE over the bright altered sequence.

Algorithm	Mean	St. Dev.	Min.	Max.
CLG	4.01	4.04	2.98	5.21
HS	7.48	7.67	4.45	10.84
BBPW	20.98	25.43	2.08	46.94

Table 4. Results for the bright altered sequence.

Brightness Difference Results: This was the noise that had the biggest impact on the results for both stereo and motion algorithms. The ones that perform the best with the original sequence are the worst in this case, except for BBPW which ranked as the second worst. SGM BT and BP are tremendously affected, while SGM MI and the dynamic programming algorithms are relatively robust to this kind of noise, see Figure 12.

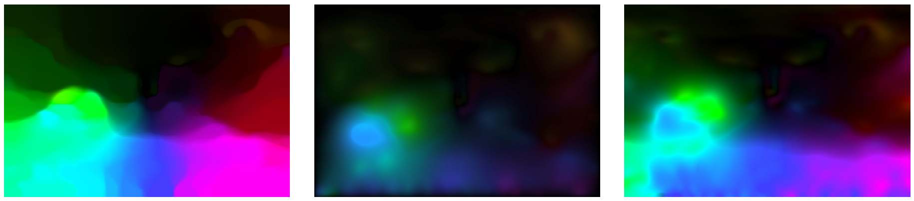
For the flow algorithms, CLG was the best and BBPW produced useless data until the difference in brightness is around 10%, see Figure 11 and Table 4. Whereas, in every frame some data can be recovered from CLG and HS. Examples of the obtained images with both algorithms over the bright altered sequence are presented in Figure 13.

Gauss Noise Results: The algorithms are very sensitive to this kind of noise too. In this case BBPW was the best for AAE and CLG for EPE, see Tables 5(a) and 5(b). For the latter metric there is a noticeable overlapping in the graphs for all of the algorithms, see Figure 15. For stereo, SGM BT is the best algorithm, and among dynamic programming algorithms, DPt is the best and DP the worst, see Figure 16. The difference between them is enough to make DPt the best overall (see Table 6(a)) dynamic programming algorithm. SGM BT and BP are relatively robust to this alteration of the images. Example of resultant images for both kind of algorithms on the Gaussian noise altered sequence are presented in Figures 17 and 18.

4.1 Algorithm Results

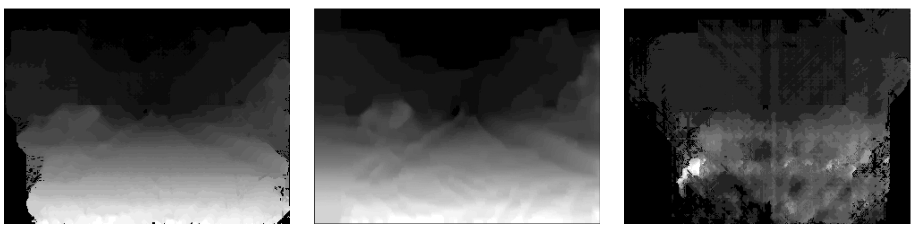
In this section we analyzed the performance of each motion algorithm. For the corresponding analysis of the stereo algorithms see [13].

BBPW: Was the best algorithm with AAE and the worst one with EPE, see Figure 19. It was the most affected algorithm by the brightness changes, where



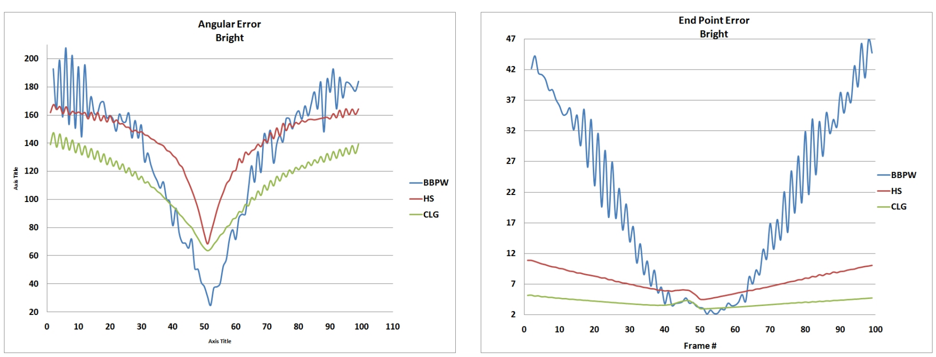
(a) BBPW (b) CLG (c) HS

Fig. 9. Results of the analysis with the motion algorithms of frames 40-41 of the blurred sequence.



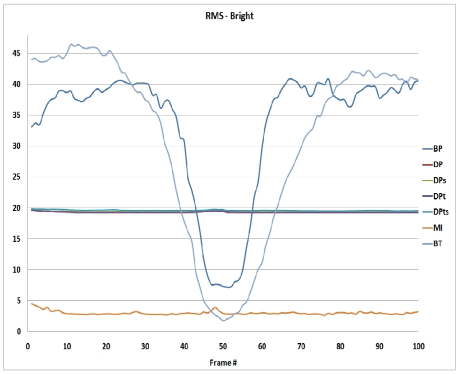
(a) SGM-BT (b) BP (c) SGM-MI

Fig. 10. Results of the three top stereo algorithm (from left to right) for image pair #40 of the blurred sequence.

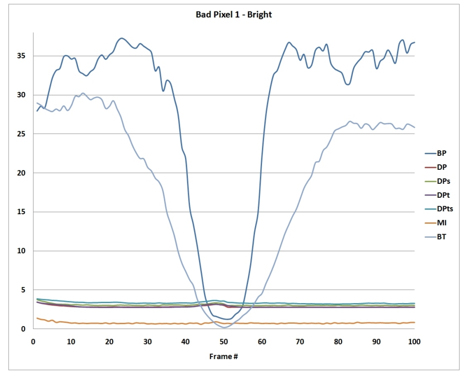


(a) AAE on bright altered sequence. (b) EPE on bright altered sequence.

Fig. 11. Results of motion algorithms over the brightness altered sequence.

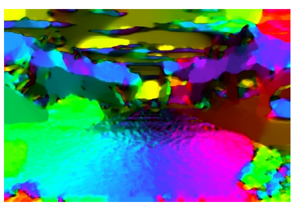


(a) RMS on brightness altered sequence.

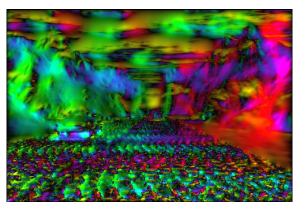


(b) BPP with $\delta = 1$ on brightness altered sequence.

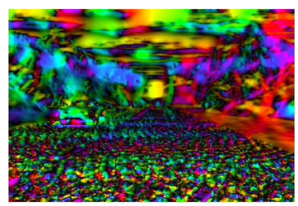
Fig. 12. Stereo results for the brightness altered sequence.



(a) BBPW

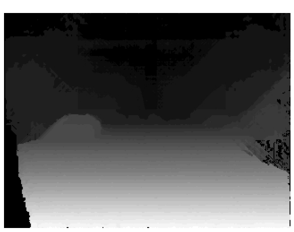


(b) CLG



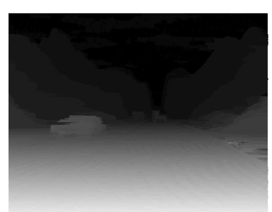
(c) HS

Fig. 13. Results of the analysis with the motion algorithms of frames 40-41 of the brightness altered sequence.



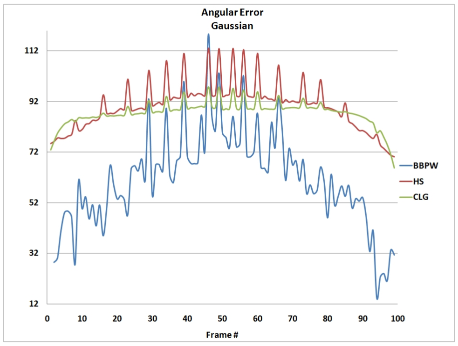
(a) SGM-MI

(b) DP

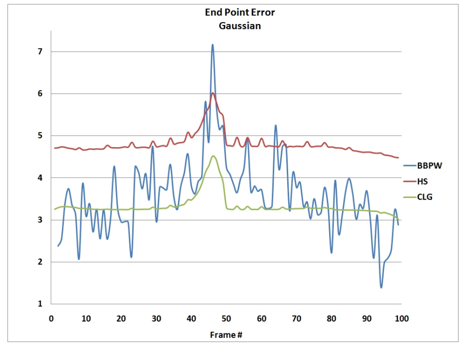


(c) SGM-MI

Fig. 14. Results of the three top stereo algorithm (from left to right) for image pair No. 40 of the brightness altered sequence.

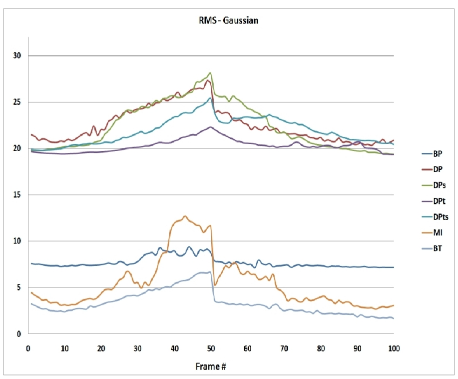


(a) AAE on Gaussian noise.

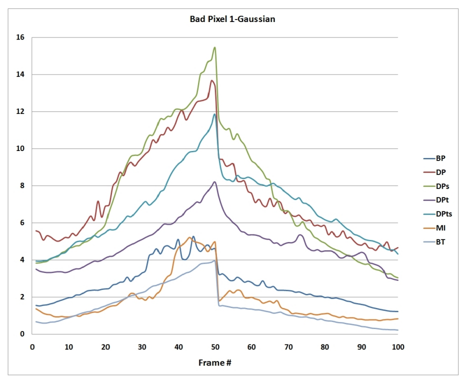


(b) EPE on Gaussian noise.

Fig. 15. Results of motion algorithms over the Gaussian noise altered sequence.

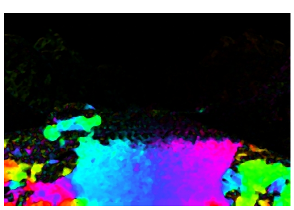


(a) RMS on Gaussian noise altered sequence.

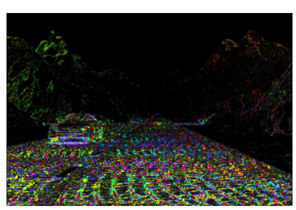


(b) BPP on Gaussian noise altered sequence.

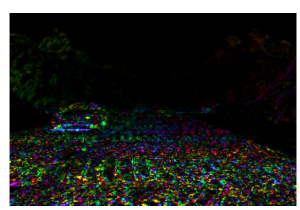
Fig. 16. Stereo results for the Gaussian noise altered sequence.



(a) BBPW



(b) CLG



(c) HS

Fig. 17. Results of the analysis with the motion algorithms of frames 40-41 of the sequence with white Gaussian noise.

(a) Results for AAE over the Gaussian noise altered sequence.

Algorithm	Mean	St. Dev.	Min.	Max.
BBPW	61.18	63.98	14.41	118.64
CLG	87.61	87.61	65.73	97.82
HS	90.00	90.48	70.12	113.03

(b) Results for EPE over the Gaussian noise altered sequence.

Algorithm	Mean	St. Dev.	Min.	Max.
CLG	3.34	3.35	3.02	4.52
BBPW	3.59	3.70	1.42	7.16
HS	4.80	4.79	4.48	6.02

Table 5. Results for the Gaussian noise altered sequence.

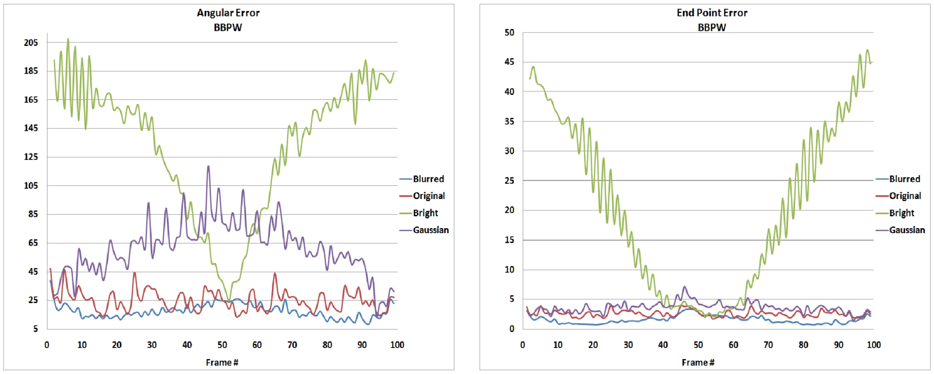
(a) SGM-BT (b) SGM-MI (c) BP

Fig. 18. Results of the three top stereo algorithm (from left to right) for image pair No. 40 of the sequence with white Gaussian noise added.

only a few resultant images can offer some useful data. It is also sensitive to the Gaussian noise. For the blurred sequence, it can be seen that there is an improvement when the blurring is between low and medium; when the blurring was maximum the results got worst. This improvement is more notorious for the EPE metric. It has to be said that when the noise magnitude of any noise is low, the difference in the error's magnitude between this algorithm and the other two is considerably large.

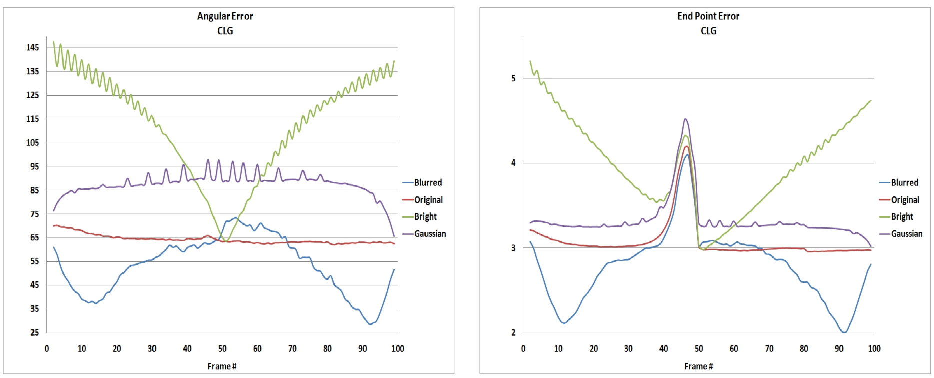
CLG: CLG is the best for the EPE metric. When the Gaussian noise is maximum, it behaves better than BBPW. The blurred sequence showed improvement too, but smaller that the other two algorithms. It was the least affected by the brightness changes. See Figure 20.

HS: HS was the worst algorithm for AAE, but the second for EPE. What is interesting with this algorithms, is that its improvement with the blurred sequence it is the most noticiable one and, even when the blurriness is maximum, there is still some improvement. See Figure 21.



(a) BBPW results for AAE. (b) BBPW results for EPE.

Fig. 19. Results of BBPW.



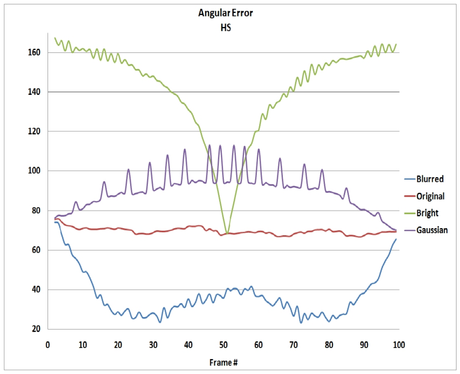
(a) CLG results for AAE. (b) CLG results for EPE.

Fig. 20. Results of CLG.

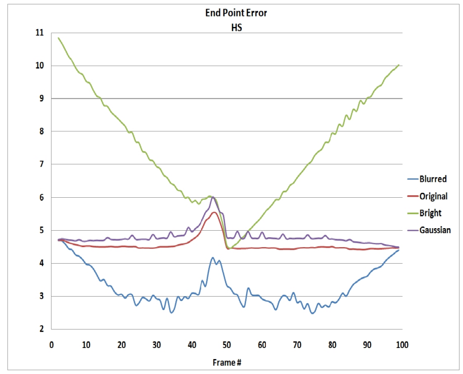
4.2 Summary

In Table 6, we present the overall statistics for the motion algorithms. For AAE, BBPW is clearly the best (with respect to the mean and standard deviation) for the sequences analyzed here. However, note that its range is the largest one, due to the very bad results that were obtained with the brightness altered sequence. The bad results with this sequence were compensated with the improvement on the blurred sequence. It is worth noting that for EPE, in the overall statistics, BBPW was the worst algorithm, once again due to results obtained with the bright altered sequence. CLG was the best for this metric. Finally, HS show an improvement with the blurred sequence, but it has also a bad performance with the brightness altered sequence.

For stereo algorithms (see Table 7) the one with better overall performance was SGM MI, with all the metrics used (for stereo algorithms) in this paper. Note that SGM BT outperform best than SGM MI in all the sequences except



(a) HS results for AAE.



(b) HS results for EPE.

Fig. 21. Results of HS.

(a) Overall motion results for AAE.

Algorithm	Mean	St. Dev.	Min.	Max.
BBPW	59.17	158.01	8.65	207.73
CLG	79.57	167.52	28.45	147.59
HS	84.72	187.83	23.16	167.36

(b) Overall stereo results for EPE.

Algorithm	Mean	SD0	Min.	Max.
CLG	3.31	6.72	2.01	5.21
HS	5.03	10.67	2.49	10.84
BBPW	7.17	25.89	0.68	46.94

Table 6. Overall results for motion algorithms over the four sequences.

in the brightness altered one, where its performance was not good at all. The DP algorithms were the worst ones for RMS, with DPt performing the best among them. It is worth saying that DPt has a better overall performance than SGM MI for the Bad Pixel metrics, see Table 7(b). BP was always below the two best algorithms, showing its worst performance in the bright sequence as with SGM BT.

5 Conclusions and future work

In this paper we presented an approach to evaluate the robustness of stereo and motion algorithms over a long synthetic sequence. In order to do this we tested several algorithms over a long synthetic sequence, which was corrupted with different kinds of noise. From our results it is clear that most of the algorithms are very sensitive to brightness differences. This has to be highlighted as changes in illumination is one of the most common problems that mobile devices have to deal with. The SGM BT stereo algorithm, whose results were the worst with this type of noise, was the best in all the other sequences. A similar behavior presented the BBPW motion algorithm. As a direct consequence of using long sequences we were able to observe that DPt represent a good option for the dynamic programming algorithms. The future work will include a wider set of noise types, more challenging sequences (real ones), a more in depth study on the

(a) Overall motion results for RMS.

Algorithm	Mean	St. Dev.	Min.	Max.
SGM MI	7.23	20.45	2.61	23.95
SGM BT	11.21	36.51	1.52	46.45
BP	15.41	39.73	6.99	40.92
DPt	20.56	41.47	19.30	29.49
DP	21.00	42.37	19.30	30.28
DPs	21.40	43.37	19.32	32.63
DPts	21.54	43.58	19.48	32.69

(b) Overall stereo results for BPP.

Algorithm	Mean	SD0	Min.	Max.
SGM-MI	3.36	11.67	0.59	15.10
DPt	4.57	11.19	2.78	15.29
DP	5.18	12.76	2.78	17.78
SGM-BT	6.07	22.16	0.17	30.23
DPs	6.07	15.75	2.93	22.40
DPts	6.21	15.41	3.21	20.87
BP	9.70	32.13	1.18	37.27

Table 7. Overall results for stereo algorithms over the four sequences.

quality metrics and a way to evaluate precisely the performance of the algorithms when there is no ground truth available.

Acknowledgments: The authors acknowledge valuable support by members of the *.enpeda..* team at Auckland university, in particular by Xi (Todd) Yang, James Milburn, and Xuan (Sean) Guo.

References

1. R. Mohan, G. Medioni, and R. Nevatia, "Stereo error detection, correction, and evaluation," *Pattern Analysis and Machine Intelligence*, vol. 11, pp. 113–120, 1989.
2. ——. Middlebury stereo and motion evaluation. [Online]. Available: `http://vision.middlebury.edu/`
3. D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. of Computer Vision*, vol. 47, pp. 7–42, 2002.
4. *.enpeda..* image sequence analysis test site (EISATS). [Online]. Available: `http://www.mi.auckland.ac.nz/EISATS/`
5. T. Vaudrey, C. Rabe, R. Klette, J. Milburn, "Differences between stereo and motion behaviour on synthetic and real-world stereo sequences," in *Proc. Int. Conf. Image and Vision Computing New Zealand*, 2008.
6. T. Vaudrey, H. Badino, and S. Gehrig, "Integrating disparity images by incorporating disparity rate," in *Robot Vision*, pp. 29–42, 2008.
7. Y. Ohta and T. Kanade, "Stereo by two-level dynamic programming," in *Proc. IJCAI*, pp. 1120–1126, 1985.
8. Z. Liu and R. Klette, "Dynamic programming stereo on real-world sequences", In *Proc. ICONIP*, LNCS, 2008.
9. P.F. Felzenszwalb and D.P. Huttenlocher, "Efficient belief propagation for early vision", *Int. J. Computer Vision*, vol. 60, pp. 261–268, 2006.
10. S. Guan, R. Klette, and Y.W. Woo, "Belief propagation for stereo analysis of night-vision sequences", in *Proc. PSIVT*, LNCS 5414, pp. 932–943, 2009.
11. H. Hirschmüller, "Accurate and efficient stereo processing by semi-global matching and mutual information", in *Proc. Computer Vision and Pattern Recognition (CVPR)*, vol. 2, pp. 807–814, 2005.
12. Open Computer Vision Library (OpenCV). Available: `http://sourceforge.net/projects/opencvlibrary/`
13. S. Morales, T. Vaudrey, and R. Klette, "An in depth robustness evaluation of stereo algorithms on long stereo sequences", *Tech. Rep. TR-30*, Available: `http://mi.auckland.ac.nz`, University of Auckland, 2008.

14. Y. Satoh and K. Skaue, "An Omnidirectional Stereo Vision-Based Smart Wheelchair", in *EURASIP Journal on Image and Video Processing*, vol. 2007, Article ID 87646, 11 pages.
15. M. Seelinger and J. D. Yoder, "Automatic Pallet Engagement by a Vision Guided Forklift", in *Proc. International Conference on Robotics and Automation (ICRA)*, pp. 4068-4073, 2005.
16. T. Brox, A. Bruhn, N. Papenberg and J. Weickert. "High Accuracy Optical Flow Estimation Based on a Theory for Warping", in *Proc. 8th European Conference on Computer Vision (ECCV)*,LNCS 5414, vol. 4 pp. 25–36, 2004.
17. S. Baker, S. Scharstein, J. P. Lewis, S. Roth, M. J. Black and R. Szelisky. "A Database and Evaluation Methodology for Optical Flow", in *Proc. IEEE 11th International Conference on Computer Vision (ICCV)*, pp. 1–8, 2007.
18. J. Barron, D. Fleet and S. Beauchemin. "Performance of Optical Flow Techniques", in *International Journal of Computer VIsion (IJCV)*, vol. 12(1), pp. 43–77, 1994.
19. A. Bruhn, J. Weickert, C. Schnrr. Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods. Updated version with errata. *Int. J. Computer Vision*, Vol. 61:211–231, 2005.