

Performance analysis of stereo reconstruction algorithms*

Néstor Morales¹, Gabriele Camellini², Mirko Felisa², Paolo Grisleri² and Paolo Zani²

Abstract—Environment mapping is one of the most critical tasks in the development of driving assistance systems and stereo vision has been widely used to accomplish it. However, there are very few datasets that allow assessing the performance of a specific method in a real world application. Most datasets have been created in controlled conditions, thus neglecting scenarios that are impossible to reproduce in a laboratory. In this paper, we present the results of the evaluation of three different dense reconstruction algorithm implementations using a number of well-known strategies that represent different trade-offs in terms of cost, set up time and accuracy. In our tests, we evaluated two variants of the Semi-Global Matching algorithm, and the Efficient Large-Scale Stereo Matching method, as well as different combinations of additional filters in order to assess their influence on the final behavior of the algorithms.

I. INTRODUCTION

Stereovision based environment mapping targeted at enabling autonomous operation of a robotic platform has been widely studied for a long time. In the last years, the computational power of hardware has greatly increased, and a number of more advanced algorithms has become viable for autonomous driving applications. A quantitative and meaningful comparison of their performance level, however, is not an easy task, mainly because of the difficulty of producing ground truth information. Older data-sets were small, and either synthetic or taken in controlled environments [1], thus effectively limiting their usefulness as indicators of the actual algorithms ability to cope with outdoor scenarios. More recently the need for suitable metrics led to the definition of improved quality measures, which will be described and used in the following. This paper will compare the performance level of some state-of-the-art stereovision-based 3D mapping algorithms, using both evaluation sets available in literature and data collected from a dedicated recording platform. One of the algorithms will also be analyzed in greater detail, and a number of variations will be tested in order to determine an improved configuration.

II. EXPERIMENTAL SETUP

Real world scenarios come with little or no data to use as ground-truth, so one solution [2], [3] is to use a high-end LIDAR (LIght Detection and Ranging) unit [4] to directly map the area surrounding the vehicle: depth measurements usually

have centimeter-level accuracy over the range 5-100 m, and produce reasonably dense maps, with up to 64 horizontal scanning planes. Another option is to exploit a prior over the data-set, such as the presence of freespace in front of a manually driven vehicle [5] to detect wrongly reconstructed points (false stereo correspondences) over an extended period of time. Finally, a virtual view synthesized from the reconstructed environment geometry can be compared with the actual data recorded by a third, suitably positioned physical camera [2], [6]. In this evaluation the algorithms have been tested using all the mentioned approaches, in order to get a better understanding of their behavior in real-world applications. Some of the tests have been carried out on data acquired using the vehicle depicted in Fig. 1, which has been equipped with a forward-looking Point Grey Bumblebee XB3-13S2C color camera with 3.5 mm optics working at a resolution of 1280×960 pixels. The imaging system is synchronized to a SICK LD-MRS-400001 4-plane LIDAR unit running at 12.5 Hz, with an angular resolution of 0.125° and a field of view of 85°, integrated in the front bull-bar. GPS and INS information are provided by a Topcon AGI-3 unit, and are used to predict the vehicle trajectory. A test sequence has been recorded in a mixed suburban and country environment. The data-set has been acquired along a 15 Km loop around the University of Parma campus surroundings; the recording session took place at around 13:14 on a sunny September day, and the scenarios include narrow country roads, small downtowns, intersections and motorways.



Fig. 1: The recording platform. Data has been collected using one of the electric vans a) which had been set up in 2010 to take part to the VisLab Intercontinental Autonomous Challenge (VIAC) [7]. b) The imaging unit is synchronized to a LIDAR c) working at a frequency of 12.5 Hz.

A. Dense LIDAR-based ground truth

As a reference, the test data-set available at [8] has been used. Ground truth for a given frame is obtained by registering 5 consecutive frames before and after the one selected and accumulating the resulting point clouds; ambiguous

*This work was supported by the Department of the Navy, Office of Naval Research (grant #N62909-12-1-7071), the project SAGENIA DPI2010-18349 and the funds from the ACIISI, cofinanced by FEDER funds (EU).

¹N. Morales is with Departamento ISAATC, Universidad de La Laguna, SPAIN. e-mail: nestor@isaatc.ull.es

²G. Camellini, M. Felisa, P. Grisleri and P. Zani are with VisLab - Dipartimento di Ingegneria dell'Informazione, Università degli Studi di Parma, ITALY. e-mail: {cgabri, felisa, grisleri, zani}@vislab.it

regions such as windows and fences are manually removed, and finally the corresponding disparity map is computed using calibration information. However, in this work scores are being computed in a slightly different manner, since the original metrics did not look completely fair. In particular:

- only non-occluded, computed pixels are being considered. The original benchmark also gives statistics after linear interpolation of missing values, with the aim of making sparse and semi-dense methods comparable to dense ones; however, such an approximation is hardly optimal, and this reflects on unfairly worsened error metrics for non-dense algorithms.
- average errors have been computed considering only the values below the endpoint error, and not all the values, in order to get a better estimate of the behavior for relevant pixels.
- statistics for each frame are being considered, not just their average over an entire sequence. To make the data easier to understand, it will be plotted in a graph with the independent variable (x-axis) representing the measured value, and the dependent one (y-axis) the percentage of frames falling below it. Better-performing algorithms are those with a lower x value for a given frame percentage (e.g. $y = 90\%$).

B. False correspondences estimation

This benchmark is an adaptation of one of the techniques described in [5]: when driving manually, a safety distance of about 1 s is usually kept from a leading vehicle; this means that a (speed-dependent) volume of free space is present at all times in front of the ego-vehicle, and any reconstructed point falling within said area must be considered as an erroneous estimate. The false correspondences percentage $m_{fc} = 100 \times N_{fc}/N$ is then the ratio of points inside the object-free volume with respect to the total number of 3D points.

C. Normalized cross correlation

The approaches introduced so far have some limitations: LIDAR-based ground truth still takes time to be produced, so it cannot be provided for large data-sets, while leading vehicle measurement can be easily performed even on long sequences, but it is an indirect performance metric, albeit a relevant one. As an alternative, the use of a third camera [6] allows to directly compare a reconstructed view with the actual images without any manual intervention. The computed disparity map is used to transform image pixels taken from the reference camera into control camera coordinates, thus effectively creating a virtual image that can be compared with that recorded by the control camera to produce a cross correlation map. The chosen metric is the Normalized Cross Correlation (NCC), computed as described in [6]. It is worth noticing that in [2] it is suggested a configuration with the reference and match camera lying 30 cm apart, and the control camera at 50 cm from the reference camera; however, the recording platform used in this work uses much

shorter distances (24 and 12 cm respectively) since it has been equipped with a pre-calibrated trinocular camera.

III. CALIBRATION

A. LIDAR to camera calibration

In order to obtain meaningful results, the LIDAR unit has been used to detect the presence of real obstacles inside the free space area. To do that, it is necessary to know the relative position between the stereo rig and the laser-scanner. This calibration procedure starts with an initial rough alignment step; after that, easily recognizable LIDAR points are manually associated to the corresponding image pixel. The accuracy of each association is constrained by several factors, such as the LIDAR angular resolution and the ambiguity of the candidates to be selected as correspondences in the image. However, with this method, a large number of samples can be quickly collected over different frames, and used together in a non-linear Maximum-Likelihood minimization framework. As a non-linear solver, the Levenberg-Marquardt approach has been chosen.

B. Camera-to-camera calibration

In order to perform the test described in Sec. II-C with the hardware setup in use the relative positioning between all of the cameras had to be computed, since the Point Grey Bumblebee XB3-13S2C device only provides combined rectification and dedistorsion look-up tables for left-right and center-right baselines.

IV. ALGORITHMS

In order to have a broad range of performance statistics, three different dense reconstruction algorithm implementations have been tested. The first two are both based on the so-called Semi-Global Matching approach (in short, SGM) first presented in [9], albeit exploiting different metrics for cost volume initialization, while the latter [10] matches sparse features in the left and right images to restrict the search range of a local window-based approach.

A. Semi-Global Matching

The Semi-Global Matching approach aims at identifying the disparity map D that minimizes the energy function

$$E(D) = E_{data}(D) + E_{smooth}(D) \quad (1)$$

with $E_{data}(D)$ representing the pixel-wise matching cost and $E_{smooth}(D)$ a smoothness constraint. In particular, the $E_{data}(D)$ term is the sum of all pixel matching costs C for the disparities of D , while the E_{smooth} term adds a small penalty P_1 to all pixels in the neighborhood of every pixel p , for which the disparity varies from p by one, and a higher penalty P_2 if the difference is greater. Global optimization of $E(D)$ is a complex task (i.e. NP-complete), and currently intractable in real time; however, good results can be obtained by applying a dynamic programming strategy.

1) *Census cost metric*: Instead of using mutual information as the pixel-wise matching function, as it is done in the original work [9], the Hamming distance of the Census transform of a 5 x 5 window cropped around each pixel has been computed, since it provides similar results [11] while reducing the overall processing burden [12]. Each position $C(p, d)$ of the cost volume is then initialized with the number of differing bits between the corresponding transformed values of the left and right images.

2) *Birchfield-Tomasi cost metric*: The freely available OpenCV SGM implementation [13] (BT-SGM in the following) uses the Birchfield-Tomasi pixel dissimilarity metric [14] to initialize the cost volume.

B. Efficient Large-Scale Stereo Matching

This method, proposed in [10] and referred to as ELAS in the following is particularly suited for handling the high disparity ranges which can arise by using large baselines or very high resolutions images. It exploits sparse, robustly matched control points to generate a 2D mesh via Delaunay triangulation, which in turn is leveraged to create a prior that is used to reduce the disparity search range for the remaining pixels. Said prior is formed by computing a piecewise linear function induced by the support point disparities and the triangulated mesh.

C. Additional filters

A number of pre- and post-processing filters, and their combinations, have been tested in order to determine which would be the most effective in improving the quality metrics discussed previously:

- Gaussian filter. A 3×3 Gaussian smoothing mask is applied to both gray-scale input images.
- Sparse Census mask. Following [15], a sparse pattern is used to compute the Census transform of the input images.
- Ternarized Census. In order to improve the amount of information about the local image structure encoded in the resulting images, the Census transform function has been modified to return three different symbols (00, 01, 11), instead of just 0, 1.
- Hamming scores aggregation. As suggested in [15], a 5×5 window centered around each pixel is used to preprocess each score in the input cube.
- Uniqueness constraint. The ratio between the first and second minima of the aggregated cost function for a given pixel is used to determine whether a match is reliable or not: higher ratios correspond to a strong minimum, which is more likely to be correct.
- Adaptive mean. An 8×8 adaptive mean filter [10] is applied over the resulting disparity map D .
- Despeckle filter. Small disparity image patches with values very different from their neighborhood are usually likely to correspond to wrong associations, so the strategy proposed in [16] is used to identify and remove them.

- Gap filter. Constant interpolation along 1D horizontal and vertical paths in the disparity image is performed in order to fill small (≤ 3 px) areas with missing disparity values [10].

Each filter has been tested individually against the Census-SGM baseline configuration, and three promising setups have been selected. Each setup has then been compared against other approaches, which also share some of the same filtering strategies, as detailed in Tab. I. For the BT-SGM and ELAS algorithms the setups suggested in [3] have been followed.

TABLE I: Algorithm configurations

	Census-SGM			BT-SGM	ELAS
	Config 1	Config 2	Config 3		
Gaussian filter	✓	✓	✓	-	-
Sparse Census mask	-	-	-	-	-
Ternarized Census	-	-	-	-	-
Hamming scores aggregation	-	-	-	-	-
Uniqueness constraint	10	20	20	10	15
Adaptive mean	✓	✓	✓	-	✓
Despeckle filter	✓	✓	✓	✓	✓
Gap filter	✓	✓	-	-	✓
Other parameters	$P_1 = 10, P_2 = 50, L/R$ check			see [17]	see [17]

V. RESULTS

In this section, full performance graphs are presented showing the results obtained for the tests performed. For the sake of brevity, the following notations will be used:

- LGT LIDAR-based ground truth evaluation (Sec. II-A).
- NFC Number of false correspondences (Sec. II-B).
- NCC Normalized cross correlation (Sec. II-C).

A. Isolated filters

LGT evaluation results for each single filter presented in Sec. IV-C are plotted in Fig. 2a and 2b. Biggest improvements can be obtained through the use of the despeckle filter; the uniqueness constraint with a strict threshold is also quite effective at removing spurious values, albeit at the cost of a reduced density. Adaptive mean consistently reduces the average reconstruction error, even if on a relatively small scale (around 0.1 px). At the level of sub-pixel error the gap filter produces worse results, which can be explained by the fact that the constant value interpolation that it performs is not accurate enough to capture pixel to pixel variations in the disparity values. Fig. 2c shows the results for the NFC test: the despeckle and uniqueness filters still show clear improvements, as it does the adaptive mean, reinforcing the idea that their combined use is likely to boost the reconstruction performance. Results for the NCC test are plotted in Fig. 2d: unfortunately, the scores obtained from the different filters are almost overlapping. This behavior is not easily explained, although some factors are likely to contribute to it:

- NCC scores are normalized with respect to the average image luminance, but local luminance variations still affect the resulting error value, so wrong reconstructions are not evenly weighted;
- the relatively small baseline in use reduces the measurable effects of reconstruction errors;

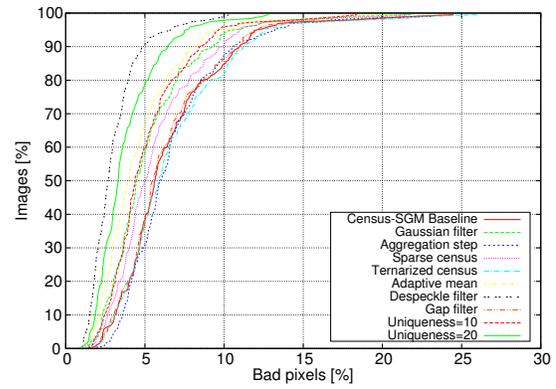
- the reconstruction quality is always quite good when using the algorithms described in Sec. IV irrespective of the filters applied, and the test scores might be dominated by other error sources, such as the calibration.

B. Composite filters

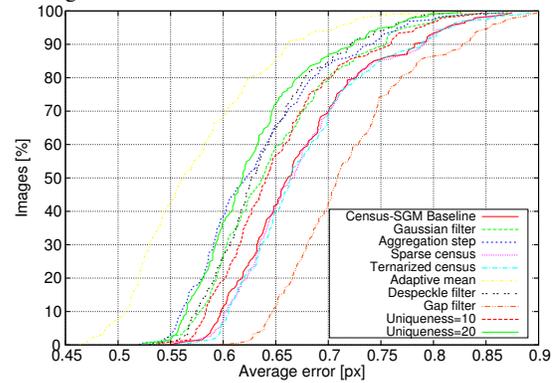
By combining different filters it is possible to obtain even better performances than when using them separately. Fig. 3a, 3b and 3c plot the results for the three Census-SGM configurations described in Tab. I under the LGT test. Looking at the 90th percentile of Fig. 3a it can be observed an improvement of around 6% in the number of pixels exceeding the endpoint error for configurations 2 and 3; the average pixels error, instead, decreases by around 0.175 px for the same two configurations (Fig. 3b). These improvements, however, come at the cost of a decreased disparity map density, as it is apparent in Fig. 3c: configuration 3, at the 90th percentile, has a density of around 58%, while configuration 2 scores better, at about 65%, which can still be considered acceptable for autonomous driving tasks; for comparison, the baseline method has a density close to 78%, with 12% bad pixels. Configurations 2 and 3 also produce the best results in the NFC test (Fig. 4a), effectively reducing the number of wrong reconstructions falling within the vehicle trajectory to a negligible amount. The NCC test, instead, seems to indicate an opposite behavior across the tested configurations (Fig. 4b), but as explained in Sec. V-A this data is likely to be very loosely related to the configuration in use.

C. Algorithms comparison

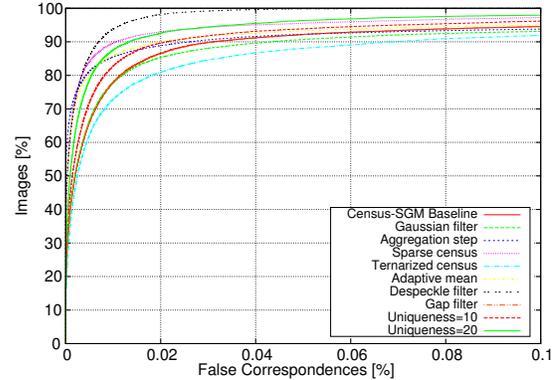
Census-SGM configuration 2 has been selected as the best compromise between reconstruction quality and map density, and the following graphs illustrate its performance compared to that of the other two approaches described in Sec. IV. LGT evaluation (Fig. 5a, 5b, and 5c) shows that the bad pixel percentage is cut by around 7.5% at the 90th percentile with respect to the baseline configuration, and by about 4.5% if compared to the BT-SGM algorithm. The average error is also reduced by 0.15 px, when using Census-SGM configuration 2, making it in line with the values obtained by ELAS. The missing pixels percentage increases to around 35%, which is 12% more than the baseline setup; however, a substantial portion of the additional unreconstructed points is due to the improved error suppression capabilities of the algorithm, and as such is expected behavior. NFC evaluation (Fig. 6a) produces results which are in line with the one obtained with the LGT test, which means that Census-SGM configuration 2 is measurably and consistently better than the alternative approaches, and as such the winning algorithm in this comparison. NCC scores for the ELAS and BT-SGM algorithms are quite close (Fig. 6b), and better than the Census-SGM baseline configuration (as expected), but the placement of Census-SGM configuration 2 looks suspicious (i.e. worse than the Census-SGM baseline configuration). For this reason, data coming from this test will have to undergo



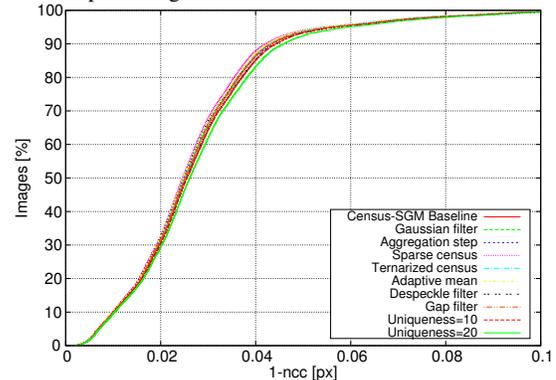
(a) Isolated filters LGT performance: bad pixels percentage.



(b) Isolated filters LGT performance: average error using LGT.

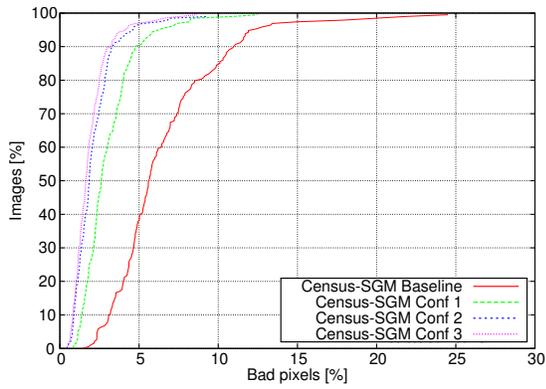


(c) Isolated filters NFC performance: false correspondences percentage.

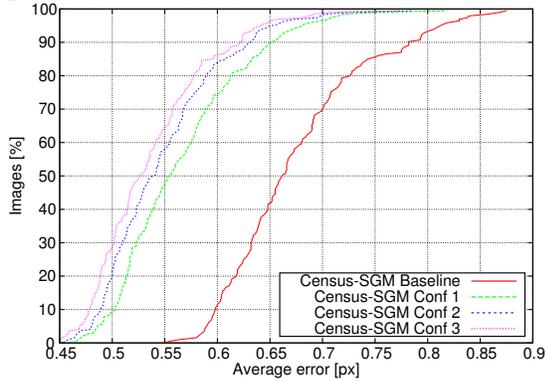


(d) Isolated filters NCC performance: 1 - normalized cross correlation.

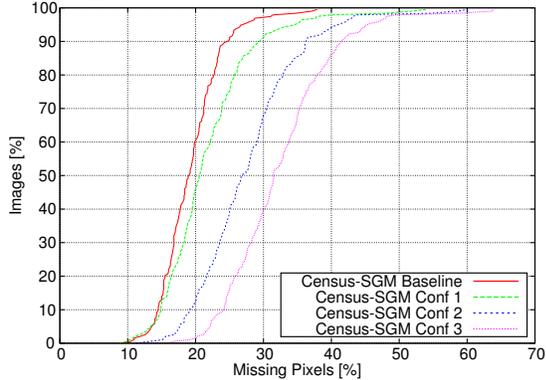
Fig. 2: Isolated filters performance.



(a) Composite filters LGT performance: bad pixels percentage.



(b) Composite filters LGT performance: average error.



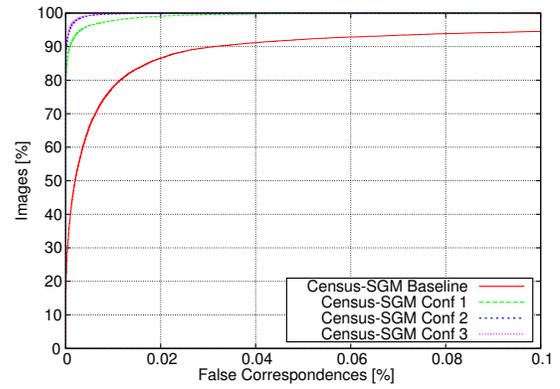
(c) Composite filters LGT performance: output density.

Fig. 3: Composite filters LGT performance.

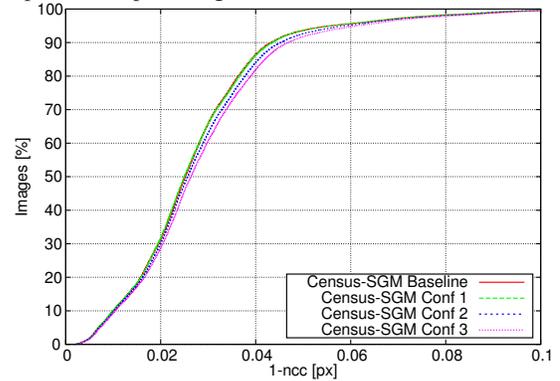
further investigation before it can be trusted as a reliable indicator of an algorithm's performance.

VI. CONCLUSIONS

The tests conducted so far have quantitatively confirmed how targeted filtering strategies can substantially reduce the amount of wrong pixels computed during stereo reconstruction, while also improving the results accuracy. In particular, the Census-SGM configuration 2 described in section IV-A.1 reduces the number of bad pixels by 7.5% at the 90th percentile, while also improving the average error by 0.15 px, making it the candidate of choice among those tested. This evaluation also served to validate the strategies employed. While LIDAR-based ground truth (see section II-A) is very



(a) Composite filters NFC performance: false correspondences percentage.



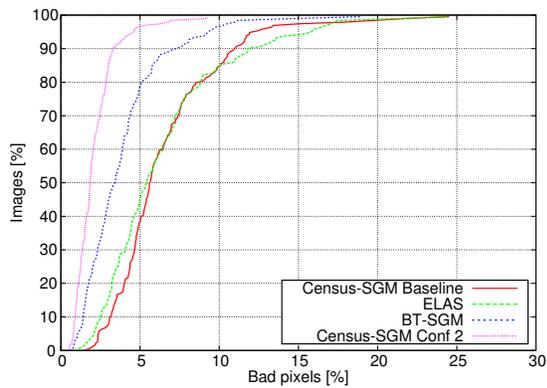
(b) Composite filters NCC performance: 1 - normalized cross correlation.

Fig. 4: Composite filters NFC and NCC performance.

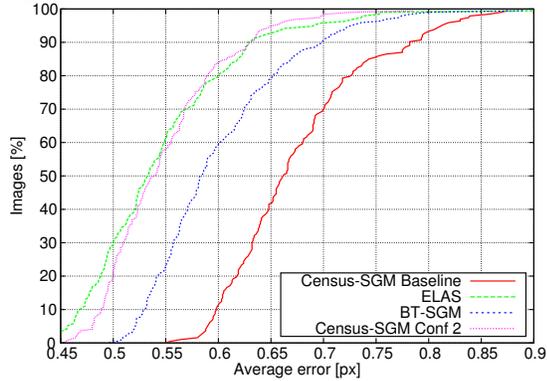
effective at producing reliable statistics, it remains quite expensive to carry out, both in terms of the equipment required and of the manual post-processing that has to be performed to produce each frame. As an alternative, the use of a prior on the vehicle movement (section II-B) can be successfully exploited to identify a portion of the wrongly reconstructed points. The advantage of this approach is that it can be effectively used to evaluate the behavior of an algorithm on big data-sets, thus covering a broad range of environmental conditions. On the other hand, the portion of space that can be checked is limited to the area in front of a moving vehicle, which often times is the most critical, but nonetheless can introduce a bias in the resulting statistics. The use of a third camera for evaluation (section II-C) is conceptually appealing, but in practice has shown to produce poor results. Further testing will be needed to assess its real effectiveness in real-world scenarios. In the future, we expect to perform more experiments including more sequences with different atmospheric conditions like other hours in a day, as well as the use of different metrics to measure the similarity between the control camera and virtual images.

REFERENCES

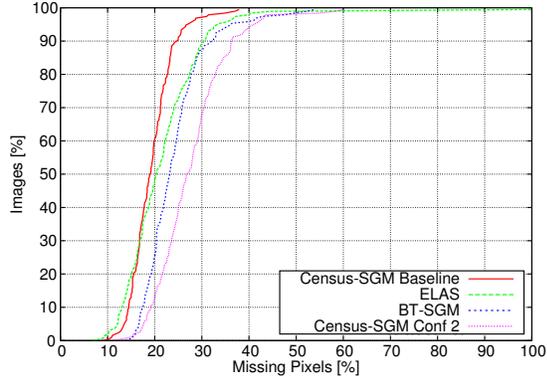
[1] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, pp. 7–42, 2002.



(a) Algorithms LGT performance: bad pixels percentage.



(b) Algorithms LGT performance: average error.



(c) Algorithms LGT performance: output density.

Fig. 5: Algorithms LGT performance.

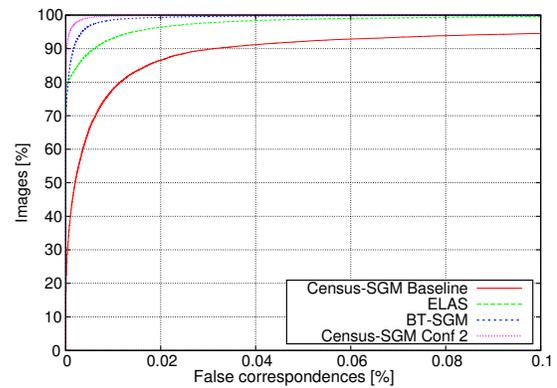
[2] S. Morales and R. Klette, "Ground truth evaluation of stereo algorithms for real world applications," in *Computer Vision ACCV 2010 Workshops*, ser. Lecture Notes in Computer Science, R. Koch and F. Huang, Eds. Springer Berlin Heidelberg, 2011, vol. 6469, pp. 152–162.

[3] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, june 2012, pp. 3354–3361.

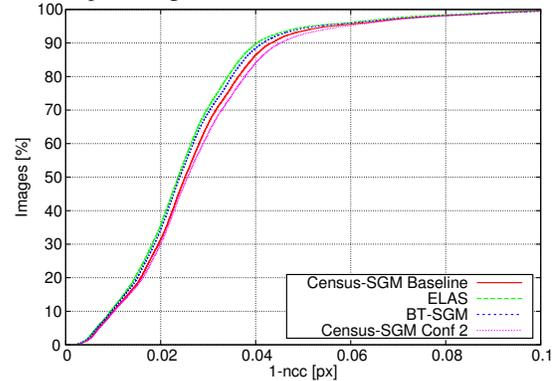
[4] <http://velodynelidar.com/lidar/hdlproducts/hdl64e.aspx>.

[5] P. Steingrube, S. Gehrig, and U. Franke, "Performance evaluation of stereo algorithms for automotive applications," in *Computer Vision Systems*, ser. Lecture Notes in Computer Science, M. Fritz, B. Schiele, and J. Piater, Eds. Springer Berlin Heidelberg, 2009, vol. 5815, pp. 285–294.

[6] S. Morales and R. Klette, "A third eye for performance evaluation in stereo sequence analysis," in *13th International Conference on Computer Analysis of Images and Patterns, CAIP'09*, 2009, pp. 1078–1086.



(a) Algorithms NFC performance: false correspondences percentage.



(b) Algorithms NCC performance: 1 - normalized cross correlation.

Fig. 6: Algorithms NFC and NCC performance.

[7] M. Bertozzi, L. Bombini, A. Broggi, M. Buzzoni, E. Cardarelli, S. Cattani, P. Cerri, S. Debatisti, R. I. Fedriga, M. Felisa, L. Gatti, A. Giacomazzo, P. Grisleri, M. C. Laghi, L. Mazzei, P. Medici, M. Panciroli, P. P. Porta, and P. Zani, "The VisLab Intercontinental Autonomous Challenge: 13,000 km, 3 months, no driver," in *Procs. 17th World Congress on ITS*, Busan, South Korea, Oct. 2010.

[8] <http://www.cvlibs.net/datasets/kitti>.

[9] H. Hirschmuller, "Accurate and efficient stereo processing by semi-global matching and mutual information," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2, june 2005, pp. 807–814 vol. 2.

[10] A. Geiger, M. Roser, and R. Urtasun, "Efficient large-scale stereo matching," in *Computer Vision ACCV 2010*, ser. Lecture Notes in Computer Science, R. Kimmel, R. Klette, and A. Sugimoto, Eds. Springer Berlin Heidelberg, 2011, vol. 6492, pp. 25–38.

[11] H. Hirschmuller and D. Scharstein, "Evaluation of stereo matching costs on images with radiometric differences," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 9, pp. 1582–1599, sept. 2009.

[12] A. Broggi, M. Buzzoni, M. Felisa, and P. Zani, "Stereo obstacle detection in challenging environments: the VIAC experience," in *Procs. IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, San Francisco, California, USA, Sept. 2011, pp. 1599–1604.

[13] <http://opencv.willowgarage.com>.

[14] S. Birchfield and C. Tomasi, "A pixel dissimilarity measure that is insensitive to image sampling," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 20, no. 4, pp. 401–406, apr 1998.

[15] C. Pantilie and S. Nedeveschi, "Sort-sgm: Subpixel optimized real-time semiglobal matching for intelligent vehicles," *Vehicular Technology, IEEE Transactions on*, vol. 61, no. 3, pp. 1032–1042, march 2012.

[16] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 2, pp. 328–341, feb. 2008.

[17] http://www.cvlibs.net/datasets/kitti/eval_stereo_flow.php.