Deep Stereo Fusion: combining multiple disparity hypotheses with deep-learning

Matteo Poggi, Stefano Mattoccia

University of Bologna
Department of Computer Science and Engineering (DISI)
Viale del Risorgimento 2, Bologna, Italy
matteo.poggi8@unibo.it, stefano.mattoccia@unibo.it

Abstract

*Stereo matching is a popular technique to infer depth from two or more images and wealth of methods have been proposed to deal with this problem. Despite these efforts, finding accurate stereo correspondences is still an open problem. The strengths and weaknesses of existing methods are often complementary and in this paper, motivated by recent trends in this field, we exploit this fact by proposing Deep Stereo Fusion, a Convolutional Neural Network capable of combining the output of multiple stereo algorithms in order to obtain more accurate result with respect to each input disparity map. Deep Stereo Fusion process a 3D features vector, encoding both spatial and cross-algorithm information, in order to select the best disparity hypothesis among those proposed by the single stereo matchers. To the best of our knowledge, our proposal is the first i) to leverage on deep learning and ii) able to predict the optimal disparity assignments by taking only as input cue the disparity maps. This second feature makes our method suitable for deployment even when other cues (*e.g.*, confidence) are not available such as when dealing with disparity maps provided by off-the-shelf 3D sensors. We thoroughly evaluate our proposal on the KITTI stereo benchmark with respect state-of-the-art in this field.*

1. Introduction

Stereo matching aims at inferring depth by determining corresponding points in images taken by two or more cameras sensing the same scene. To this end several approaches have been proposed and a quite outdated, yet exhaustive, reviewed and evaluation on a small and unrealistic dataset was proposed in [29] . Despite the research efforts in this field, with the introduction of more challenging datasets such as KITTI [6, 7, 24] and Middlebury 2014 [30] it is clear that even the most accurate approaches such as [41] are still far from correctly solving the correspondence problem.

Most stereo algorithms rely on a set of parameters or heuristics which perform very well in particular circumstances but yield poor results in others. A typical example is the size of the aggregation window used by local methods, which should be large when dealing with smooth frontal-parallel surfaces and smaller near depth discontinuities or slanted surfaces. Other methods, such as Semi Global Matching (SGM) [12] perform pretty well in smooth and slanted areas but may lead to artifacts near depth discontinuities. These observations lead some researchers [33, 25] to argue that the overall disparity accuracy can be improved exploiting redundancy in the input data by means of decision trees.

In this paper we follow the same intuition but following a completely different strategy. We propose Deep Stereo Fusion (DSF), a novel end-to-end methodology to predict a more reliable disparity map taking as input the output of multiple Stereo Matchers (SMs). Differently from [33], based on explicit features extraction from input data (*e.g.*, confidence, matching costs, etc), DSF relies on deep learning, deploying a Convolutional Neural Network (CNN), aimed at processing only the disparity maps provided by multiple SMs. This is carried out by tackling fusion as a multi-labeling classification problem, which enables to design a single classifier capable to predict, according to the the input sample, the reliability of each matcher. The outcome is a *choice map*, shown in Figure 1, encoding for each pixel which SM is selected at each location.

This strategy enables an elegant end-to-end training and testing procedure, conversely to other approaches which leverage on multiple classifiers (*e.g.*, one per SM), requiring stand-alone training procedures. Moreover, it leads to a significantly faster response time. We evaluate DSF on the KITTI 2012 dataset [6, 7], comparing our results with state-of-the-art approach represented by Spyropoulos and Mor-

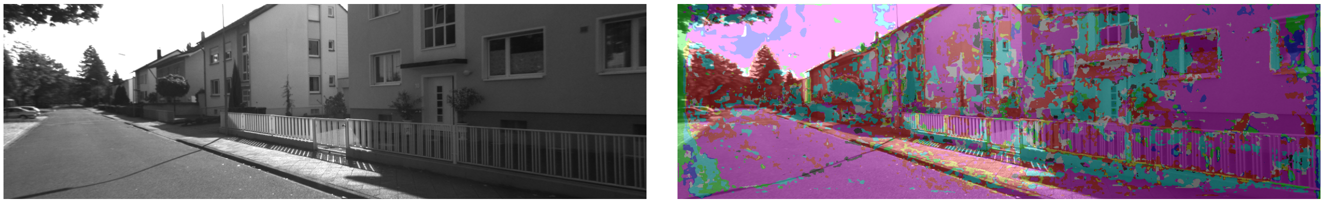
(a) (b)

Figure 1. Overview of Deep Stereo Fusion. (a) reference image from a KITTI 2012 [6, 7] stereo pair (000117), (b) *choice map*, each color encodes a different SM: for each point is depicted the one selected by the framework [best viewed with colors].

dohai [33].

2. Related Work

According to [29], stereo algorithms can be broadly classified in *local* and *global* according to the steps performed during their execution, which are matching cost computation, cost aggregation, disparity computation/optimization and disparity refinement. Local methods [36, 13, 14, 5], usually perform the first two steps while global methods [34] mostly focus on the first and the third. Local approaches usually are faster, but are more prone to errors especially on challenging datasets such as the KITTI [6, 7, 24] and Middlebury 2014 [30]. On the other hand, by minimizing and energy term on the whole image exploiting appropriate approximated strategies [34], global approaches provide in general more accurate results. A good compromise between the two categories is the popular SGM algorithm [12] that, solving multiple disparity optimization problems independently, provides a good trade-off between accuracy and speed.

Machine learning techniques, initially adopted by the computer vision community for high level tasks such as *scene understanding*, have been more recently deployed also for tackling low-level vision problems such as stereo or confidence estimation. Some early works focused on the estimation of parameters to better tune an algorithm such in the case of MRF based methods [38, 42] while other for feature selection in stereo [21]. A first attempt to detect correct and wrong matches leveraging on machine learning were proposed in [4, 28]. Others proposed the joint use of multiple confidence measures [15], by means of random forest classifiers, to obtain a more effective index of correctness for disparity assignment. In [26, 10, 32, 27] similar strategies were focused at improving the accuracy of disparity maps exploiting learned confidence measures. In particular, Spyropoulos et al. [32] detected a set of very confident pixels, referred to as *ground control points*, and inferred the disparity on the whole image leveraging on these seeds, Park and Yoon [26] modulated the initial matching cost curves before running aggregation/optimization steps according to an estimated confidence, In [27] we identify the reliability of each scanline of the SGM algorithm weighting in the overall sum their contribution accordingly.

The idea of exploiting redundancy by combining different algorithms has been already applied to different fields. Zhu et al. [43] fit regression models to local image areas, starting from the assumption that a single one is not suited for all pixels. Some approaches modeled it by a multi-label classification problem as well, like [17] which assigns to each match the probability to belong to three categories and [20] addressing both semantic segmentation and 3D reconstruction. Even more related to the task we are tackling, some works on stereo [18, 25, 33] and optical flow [22] deploy frameworks able to choose the best assignment starting from several hypothesis provided by different algorithms. Some of these, for instance [22], do not take into account any kind of mutual information between the input cues. On the other hand, others [33] exploit this fact enabling their classifier to improve the error rate.

Deep learning techniques recently started to spread on low level vision tasks too. In [39], Zagoruyko and Komodakis reported a complete study on how to learn directly from image data a general similarity function by exploiting CNN architectures. Specifically, they used 2-channels, *siamese* and *pseudo-siamese* models, reporting results related to stereo matching as a particular case of image matching. Zbontar and LeCunn, proposed [40, 41] an effective methodology for matching cost computation relying on a CNN, able to rank at the top of KITTI [6, 7, 24] and Middlebury [30] when post-processed by a state-of-the-art stereo pipeline based on SGM and a local cost aggregation approach. Mayer et al. [23] designed a end-to-end fully-convolutional network, processing full-resolution stereo pairs and inferring depth without any operation typically performed by stereo algorithms. Deep architectures usually require huge amount of data for training: popular stereo datasets [6, 24, 30] provides enough samples when designing networks with a relatively small perceptive field (*i.e.*, dimension of the images processed during the training phase), but they provide an almost insignificant number of training samples for architectures with fully-resolution perceptive fields such as *i.e.* [23]. Some authors dealt with this issue by proposing a data-augmentation process [3] leveraging on multiple view points and contradictions between

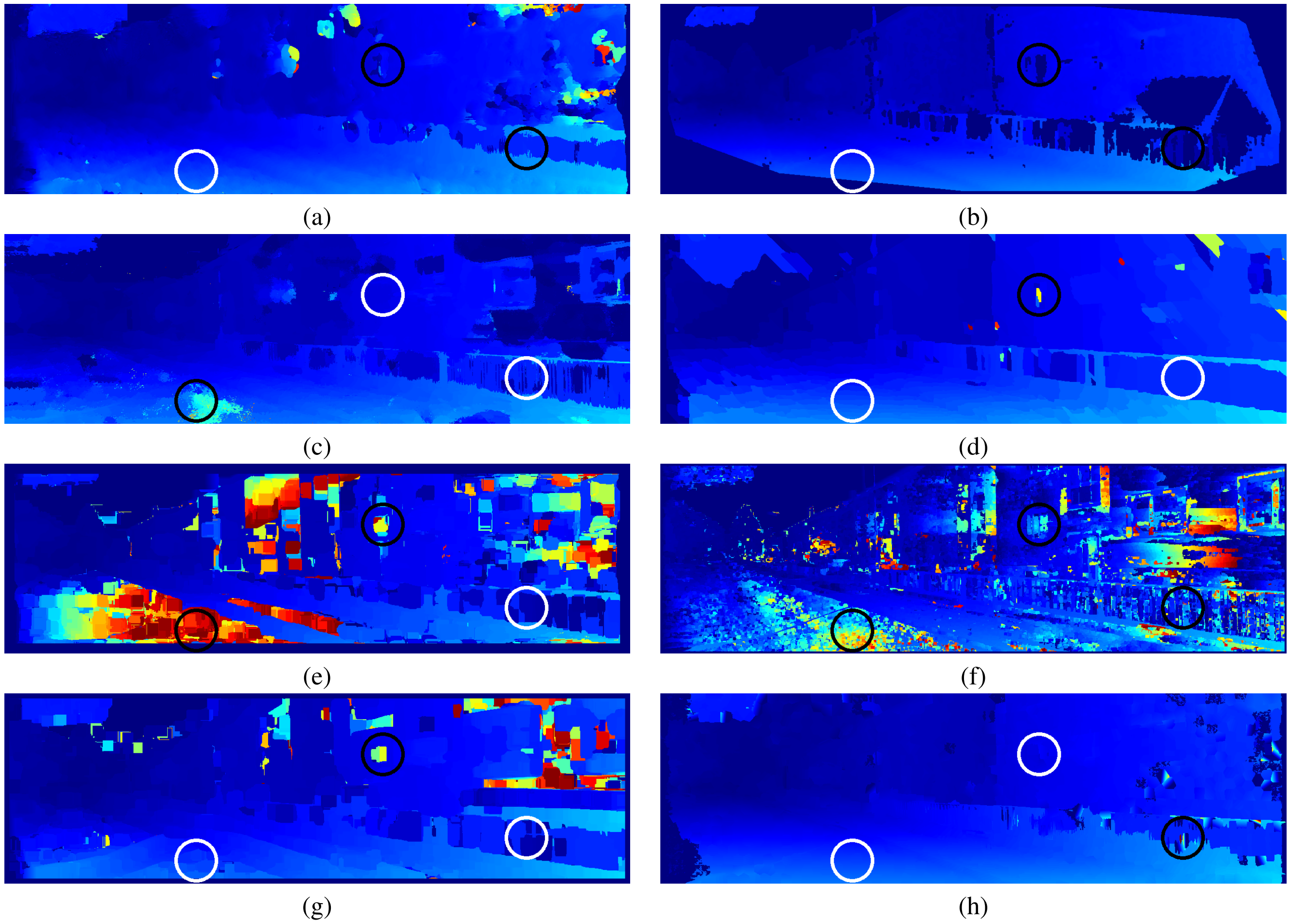

Figure 2. Disparity maps obtained from a set of different SMs on the KITTI stereo pair 000117, shown in Figure 1. We can notice as the same local area presents artifacts on the output of some matchers (black circles), while it appears to be correct on others (white circle). (a) DAISY, (b) ELAS, (c) FCVF, (d) MRF, (e) SH-SOB21, (f) SH-SSD5, (g) SH-ZNCC21, (h) SUPER-rSGM5. Detailed description of the matchers in Section 3.2.

multiple depth maps, or producing synthetic datasets [23] large enough to run an end-to-end training of a deep architecture. Finally, CNNs have been deployed in multi-label classification tasks and Xi'an et al. [9] deployed a CNN for cross-domain action unit detection, Wei et al. [37] propose a methodology to train a network on single-label samples by choosing a proper loss function and fine-tune on multi-label examples, Kurata et al. [19] propose to treat some of the neurons in the final hidden layer as dedicated neurons for each pattern of label co-occurrence.

3. Deep Stereo Fusion

In this section we introduce the DSF framework that, given a set M of SMs, aims at combining multiple input disparity maps D_k, $k \in M$ to obtain a more accurate map D_F. By deeply analyzing this problem we decided to model it as a multi-label classification problem driven by two main assumption:

- Each SM compute disparity assignments according to different cues. Usually, different behaviors can be observed locally on disparity maps by changing the stereo algorithm (*e.g.*, near depth discontinuities, on low-textured areas, etc), as depicted in Figure 2. A framework aimed at merging different SMs should be able to distinguish, in any circumstance, the best assignment according to local properties of the input disparity maps. The different disparity maps can be seen as different *features* provided to the merging classifier.

- Choosing among a pool of SMs can be casted within a classification problem: given a sample made of $m = |M|$ features, the framework choose the category (*i.e.*, most accurate SM) it belongs to. Moreover, for a given pixel one or more matchers could possibly vote for the same, correct disparity assignment, leading to a multi-label classification problem.

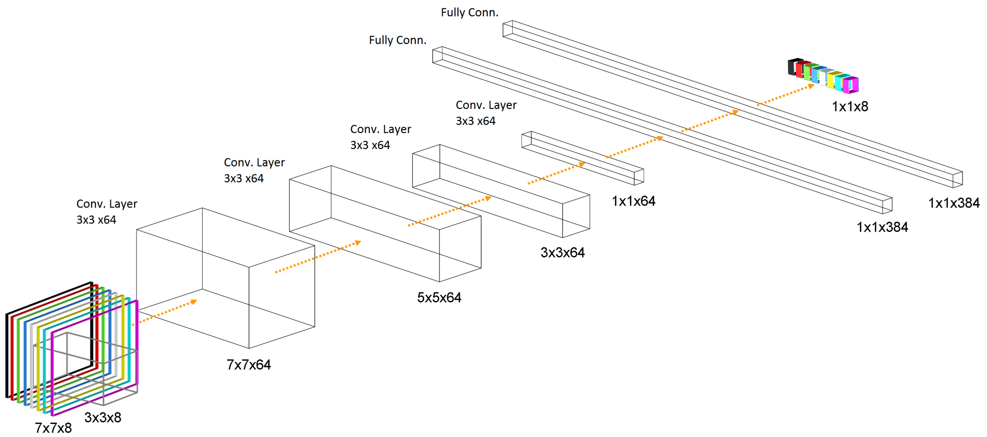

Figure 3. Architecture of the DSF. Input MSI, containing disparity maps from the eight matchers, is forwarded to four convolutional layer and, then, to two fully-connected layers, each of them followed by ReLU activators. The output vector encodes a score for each matcher. For a given input, the highest score locates the best SM.

According to these assumption and to the recent achievements in this field [23, 3, 40, 41], we train, on a large dataset with ground-truth, a deep architecture aimed at dealing with the outlined problem. For each pixel, we extract m square patches centered on the D_k, $k \in M$, disparity maps. These data are collected inside a 3D tensor, a Matcher Space Image (MSI). The DSF is trained on a large set of MSIs in order to distinguish the best matcher on the different samples and, thus, it provides a set of m scores. The chosen matcher will be the one with the highest score. The features representation encoded by MSI allows us for a joint processing of data provided by the different matchers, similarly to what Spyropoulos et al. achieved by an explicit computation of features encoding agreement among the SMs, but deployed within a single classifier instead of using a single classifier per matcher [33].

We introduce the architecture of our network in section 3.1, then we report details about our experimental evaluation, by defining the same set M of 8 matchers in Section 3.2 in order to compare our proposal with [33].

Section 3.3 provides details about the training phase and finally, in Section 4, we thoroughly compare our proposal [33] on the KITTI 2012 [6, 7] dataset.

3.1. Proposed architecture

Figure 3 shows the architecture of DSF, organized as a m-channels network. The input to the network is a MSI of dimension $N \times N \times m$, with N the side of squared patch extracted from the input disparity maps provided by the m methods. According to state-of-the-art methodology deployed for stereo, for our experiments we tuned N equal to 9. This means that SDF has a perceptive field of 9×9, from which it will determine, for the central pixel, the optimal disparity assignment. This quite small perceptive field allows us for generating a large amount of training samples from the available datasets for stereo [6, 7, 24, 30] without requiring synthetic data. the DSF then extracts a large number of features, by deploying four convolutional layers. Each layer is made of F convolutional kernels of size 3×3, each one followed by a Rectifier Linear Unit (ReLU).

$$ReLU(x) = \begin{cases} 0, & \text{if } x < 0 \\ x, & \text{otherwise} \end{cases} \tag{1}$$

Being no padding or stride applied, these four layers lead to a 1D output tensor, more precisely of size F. This feature vector is then forwarded to two fully-connected layers followed by ReLU. Finally, a classification layer is in charge of predicting which of the considered matchers propose the best disparity assignment, by a layer made of m neurons providing a 1D prediction vector C containing m values. The optimal disparity assignment for the central pixel inside the perceptive field is assigned as

$$D_F(x, y) = D_w(x, y) \tag{2}$$

with w index of (one of) the matcher(s) achieving the highest score from the prediction layer.

Recent advances in deep learning introduced the concept of fully-convolutional neural network, an architecture on which traditional fully-connected layers are absent and the whole classifier consists only on convolution and sampling operation. Any model deploying fully-connected layer is equivalent to a a fully-convolutional network by replacing each fully-connected layer with a convolutional one made of 1×1 kernels, as many as the number of neurons of the replaced layer, with the same weights and biases. The main benefit of this approach is to model a size-independent classifier, able to process data (images) of any size without cropping or resizing the input. This allows our framework, as well as other works [40, 41, 2], to greatly speed-up run time execution, enabling for a single forward of a full resolution MSI instead of $w \times h$ forwards of 9×9 cropped data. The absence of pooling operation inside DSF leads to an output of size $(w-8) \times (h-8) \times m$, by applying a *0-padding* of size 4 around the whole input dimension w and h enables to obtain a 3D prediction tensor of size $w \times h \times m$, reducing the run time required by the DSF framework from several minutes to few seconds.

3.2. Combined Stereo Matchers

As SMs for the experimental evaluation we chose the same pool M of $m = 8$ of stereo algorithms deployed in [33] in order to be able to directly compare with it.

- DAISY: an approach based on a local descriptor aimed at wide baseline stereo matching [35]
- ELAS: Efficient LArge-Scale [8] stereo matching detects an initial set of reliable disparity assignments and fills remaining one with an appropriate triangulation
- FCVF: local method based on edge-preserving filtering of cost volume [14, 5] by means of the guided filter [11]
- MRF: global method expressed within a Markov Random Field framework [16]. Matching cost is Normalized Cross Correlation on 5×5 windows and smoothness penalty is modulated according to intensity difference between neighboring pixels
- SH-SOB21: Shiftable Window local aggregation on 21×21 patches. Matching cost is sum of absolute differences (SAD) of responses to vertical edge (*e.g.*, Sobel filter on x direction)
- SH-SSD5: Shiftable Window aggregation on 5×5 boxes. Initial costs processed as sum of squared differences (SSD) of color intensities
- SH-ZNCC21: Shiftable Window aggregation on 21×21 boxes. Initial costs processed as Zero-Mean Cross-Correlation (ZNCC) on 21×21 patches
- SUPER-rSGM5: SGM [12] variant proposed by Spangerberg et al. [31] as. Input images are census transformed on 5×5 patches. The output of the algorithm is further enhanced exploiting superpixels as described in [33], segmenting left image into SLIC superpixels [1] and fitting a plane for each segment with RANSAC.

3.3. Training procedure

In our experiments, we trained the DSF framework on the first 50 frames of the the KITTI 2012 dataset [6, 7] on cropped samples centered on pixels with available ground-truth values (approximatively $\frac{1}{3}$ of the overall pixels). This strategy provides more than 6.5 million MSIs of dimension $9 \times 9 \times 8$ for the training set. For each sample, a label vector of dimension 8 is assigned, encoding the correctness of a disparity assignment for each of the 8 SMs belonging to set M. If a given matcher provides a disparity assignment which differs from the ground-truth value for more than 3, it is labeled as wrong assignment, encoded as '0' in the label vector, '1' otherwise. Differently from the strategy adopted in [37], we directly trained our model on multi-label samples, in order to the reduce the amount of *single-label samples* (*i.e.*, pixels having only a single matcher proposing the correct assignment) due to the high overlapping between the correct matches predicted by the different SMs. Otherwise, the amount of training data would be drastically reduced. We tuned DSF hyper-parameters, achieving the best results with $F = 64$ kernels for each convolutional layer and 384 neurons for the fully-connected layers (thus, 384 kernels 1×1 in the fully-convolutional model). During the training phase, we followed the Stochastic Gradient Descent (SGD). We optimized the Binary Cross Entropy loss function (BCE), extended to the multi-label classification problem as in [9, 37, 19], between output o of the network and label t on each sample i of the mini-batch B (3)

$$BCE(o,t) = -\frac{1}{n}\sum_{i\in B}\sum_{k\in M}\Bigg(t[i][k]\log\left(o[i][k]\right) + \left(1-t[i][k]\right)\log\left(1-o[i][k]\right)\Bigg) \tag{3}$$

by adding a sigmoid function S(x) (4) as final layer of the network

$$S(x) = \frac{1}{1+e^{-x}} \tag{4}$$

We carried out 60 training *epochs*, with an initial *learning rate* of 0.003, decreased by a factor 10 after the 10^{th} epoch and after the 30^{th}, leading to a final learning rate of 3×10^{-5} for the final 30 epochs and a *momentum* of 0.9, inspired by [41] and confirmed by our experiments. The size of each mini-batch B was 128. The whole training procedure, carried out on a i7 4720HQ CPU, took approximatively 4 days. To speed-up the training procedure, DSF was first designed with fully-connected layers, which appears to be faster with respect to 1×1 convolutional layers during this phase. Once the network was trained, we replaced fully-connected layers with fully-convolutional ones.

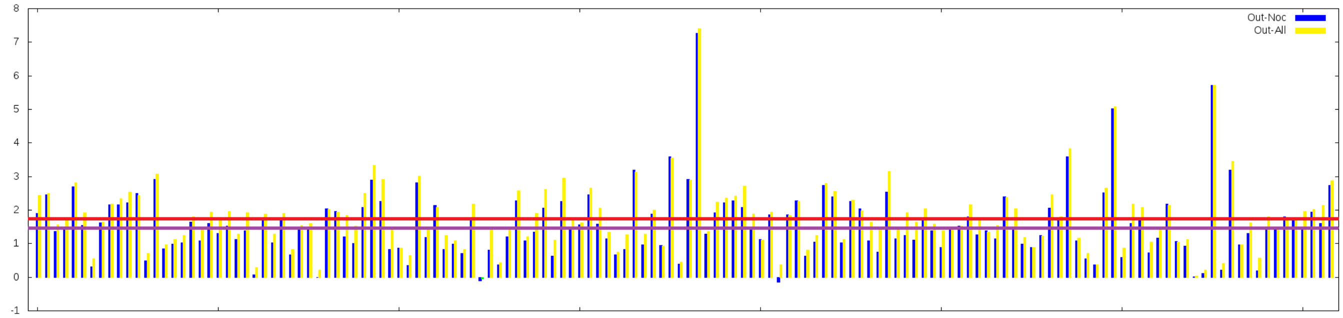

Figure 4. Absolute improvement of the error rate with respect to the most accurate matcher in M, SUPER-rSGM5, yielded by DSF on non-occluded (blue) and all pixels (yellow) of the test set (*i.e.*, KITTI 2012 frames from 000050 to 000193). The network is able to improve the error rate on all the considered frames except 000132 when considering non-occluded pixels and 000099 on both cases. DSF achieve an absolute improvement on the test set of 1.51% on non-occluded pixels (red line) and 1.75% on all pixels (purple line).

Algorithm	Out-Noc	Out-All
DAISY	10.88%	12.86%
ELAS	19.90%	21.68%
FCVF	21.59%	22.46%
MRF	10.60%	12.58%
SH-SOB21	43.64%	44.86%
SH-SSD5	55.08%	56.04%
SH-ZNCC21	30.71%	29.20%
SUPER-rSGM5	7.85%	9.90%
DSF	6.34%	8.14%

Table 1. Error rate achieved by the 8 SMs and DSF on our test set (*i.e.*, KITTI 2012 frames from 000050 to 000193) on both non-occluded (Out-Noc) and all pixels (Out-All) with ground-truth available.

Algorithm	% of total pixels
DAISY	1.04%
ELAS	21.58%
FCVF	6.76%
MRF	3.31%
SH-SOB21	7.73%
SH-SSD5	3.84%
SH-ZNCC21	17.88%
SUPER-rSGM5	37.85%
Total	100.00%

Table 2. Average occurrence rate of each matcher selected as winner by DSF on KITTI 2012 frames from 000050 to 000193.

4. Experimental results

We evaluated the proposed DSF framework[1] on the remaining 143 frames of the KITTI 2012 dataset not used during the training phase by computing: the error rate of the merged disparity map over all pixel with available ground-truth (*i.e.*, *disp_occ* data provided by KITTI 2012) and non-occluded areas (*i.e.*, *disp_noc* data provided by KITTI 2012), reported, respectively, as *Out-All* and *Out-Noc* rates. Then, we compared our results with the proposal of Spyropoulos et al. [33] on the same latest 97 frames of KITTI 2012 dataset (in [33] the training set was made of the first 97 frames out of 194).

Figure 4 plots the difference in terms of error rate for Out-Noc and Out-All between the most accurate SM in the M set, which is SUPER-rSGM5, and the output provided by DSF on the KITTI 2012 dataset, excluding the first 50 stereo pairs involved in training procedure. Positive values stands for a reduction of the error rate carried out by our proposal. Except two cases, which are stereo pair 000099 for both Out-Noc and Out-All, 000132 for Out-Noc, DSF is able to effectively merge the 8 matchers and outperforms SUPER-rSGM5, with an absolute average error rate reduction of 1.51% Out-Noc and 1.75% Out-All. The error rate over the whole test set for SUPER-rSGM5 is 7.85% Out-Noc and 9.90% Out-All, while DSF achieve respectively 6.34% and 8.14%, with a relative improvement of 19.23% and 17.7% respectively. Table 1 shows average error rates for all the 8 SMs, as well for DSF.

Figure 5 plots the occurrence rate of the matcher selected as winner by DSF. Each bar of the histogram represents a single stereo pairs from our test set, the different colors encode the 8 combined matchers according to the legend reported in the figure. We can notice how the most accurate algorithm, SUPER-rSGM5, is chosen most of the times, as we could expect, while two of the most accurate methods after SUPER-rSGM5 are not frequently selected. This fact is not necessarily inconsistent with the nature of the problem: a large subset of pixels which are correctly assigned by multiple SMs decreases the possibility of a matcher to be dominant with respect to the others. This is a direct consequence of the multi-label classification task we modeled to deal with the problem. However, the reported error rates support the multi-labeling assumption adopted. Table 2 summarizes these results, reporting the occurrence rate over our

[1] Source code available on the authors' website

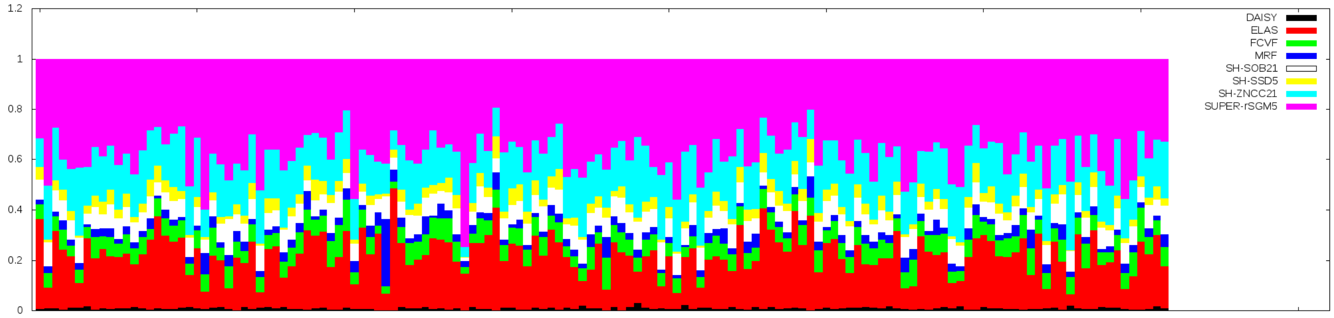

Figure 5. Occurrence rate of the matcher selected as winner by DSF on each image of our test set (*i.e*., KITTI 2012 frames from 000050 to 000193). SUPER-rSGM5 (purple) is the most selected algorithm, followed by ELAS (red) and SH-ZNCC21 (cyan). SH-SOB21 (white) and FCVF (green) gives a minor contribution to disparity selection, followed by SH-SSD5 (yellow) and MRF (blue). Finally, DAISY (black) is seldom selected as winner.

entire test set, confirming the trend previously highlighted with the histogram.

Then, we computed the error rate on the last 97 stereo pairs from KITTI (*i.e*., from 000097 to 000193) in order to compare our proposal with the N8 framework proposed in [33]. We do not apply calibration nor post-processing reported in their paper to compare the raw predictive capability of the two classifiers. Table 3 reports the result of this comparison. While the N8 ensemble classifiers[33] performs better when processing non-occluded pixels only, DSF slightly outperforms it when considering all pixels. Therefore, the two methods can be considered almost equivalent, but DSF performs better when dealing with occluded areas. In fact, Table 4 reports the error rate, on the same testing set, restricted to occluded areas only. N8 achieves a 96.28% error rate, while DSF 87.46%, outperforming it with an absolute error reduction of 8.82%. Our proposal does not rely on explicit features extraction, while N8 requires a set of hand.crafted indexes encoding matchers agreement and the Left-Right Consistency check (LRC) which might be available with the disparity maps. This latter fact makes our method suited for fusing disparity maps provided by any kind of stereo sensor, including out-of-the-box device. The LRC features available may also be responsible of the lower accuracy achieved by N8 on occluded area, being it a strong information driven by depth discontinuities. According to [33], given the full disparity maps obtained by the 8 SMs, the time required to test the whole testing set (97 stereo pairs) is more than 3 hours as verified on the same CPU, leading to an average 100+ seconds per stereo pair. This means that each of the 8 classifiers takes about 12-13 seconds. On the other hand, the fully-convolutional nature of DSF, on the same CPU, makes out method much faster requiring about 10 seconds for each stereo pair (0.65 s on a Titan X GPU). Finally, Figure 6 depicts intermediate and final results provided by DSF, which are the single score maps related to each algorithm (b-i) and the final choice map (j).

Algorithm	SUPER-rSGM5	N8 [33]	DSF
Out-Noc	8.06%	**6.21%**	6.37%
Out-All	10.17%	8.21%	**8.18%**

Table 3. Error rate achieved on the last 97 stereo pairs from KITTI 2012 dataset (to compare with the evaluation reported in [33] on all pixels with provided ground-truth. DSF outperforms the most accurate matcher and shows to be almost equivalent to N8 [33] deploying a single classifier instead of 8 [33]. The execution time of DSF with respect to N8 is reduced by a factor 10.

Algorithm	N8 [33]	DSF
Occlusions	96.28%	**87.46%**

Table 4. Error rate achieved on the last 97 stereo pairs from KITTI 2012 dataset, considering only occluded pixels. DSF performs better than N8 [33].

5. Conclusions

In this paper, we introduced Dense Stereo Fusion, a novel framework aimed at combining the output of several stereo algorithms. Our proposal allows for an elegant end-to-end training and testing of a single classifier, conversely to other approaches deploying multiple classifiers [33]. Moreover, it enables a much lower running time with respect to the same method. Experimental results confirm that DSF is able to outperform all the combined matchers and is almost equivalent, in terms of accuracy, to state-of-the-art framework proposed by Spyropoulos et al. [33]. Nevertheless, our network clearly outperforms it on occluded pixels, proving to be more robust in such critical areas.

Acknowledgments The authors would like to acknowledge with thanks Aristotle Spyropoulos and Philippos Mordohai for providing disparity maps deployed for training and evaluation.

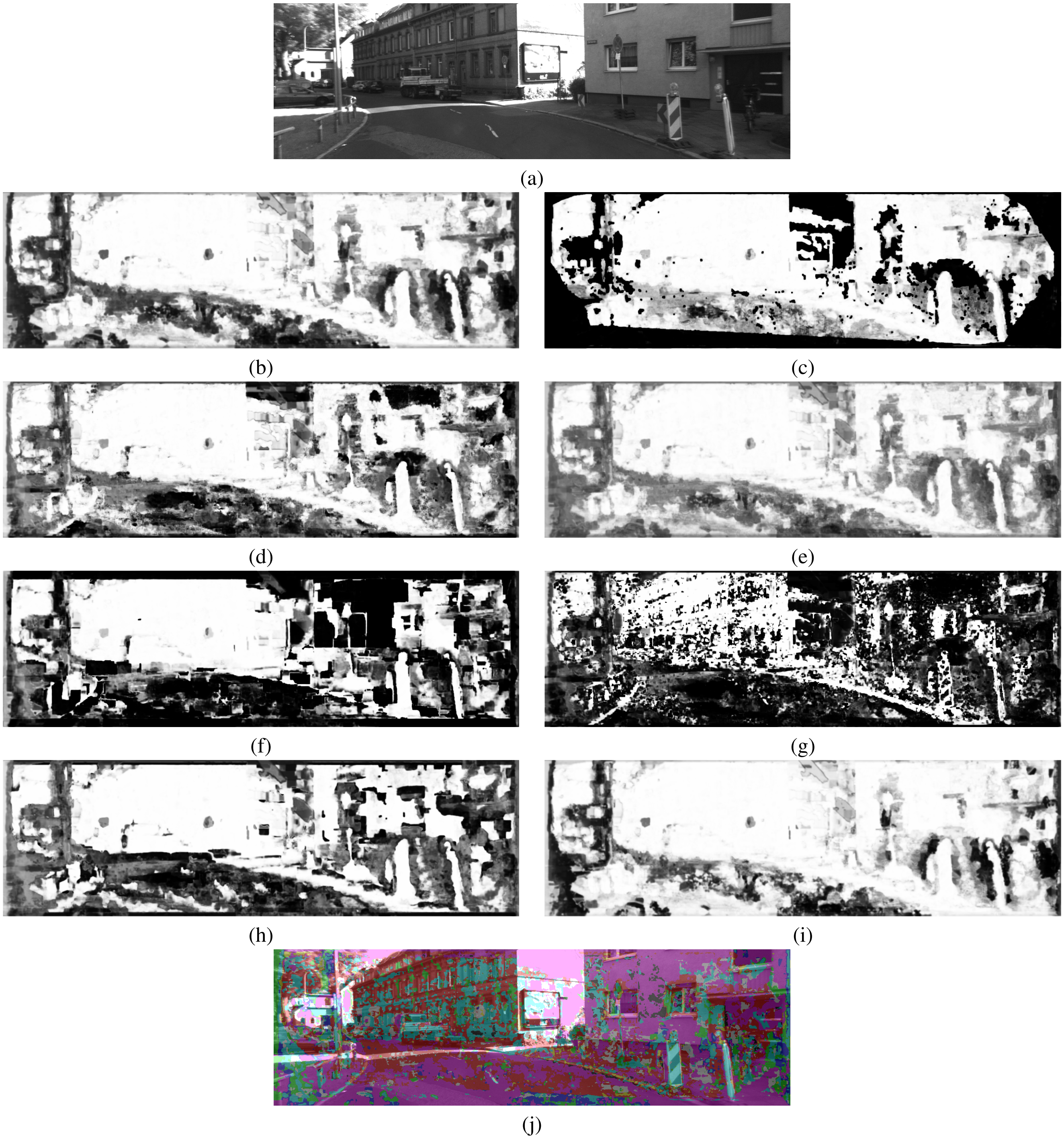
(a)
(b)
(c)
(d)
(e)
(f)
(g)
(h)
(i)
(j)

Figure 6. DSF pipeline highlighting intermediate results. (a) reference image for KITTI stereo pair 000123, (b-i) scores assigned by DSF to the different matchers (in order: DAISY, ELAS, FCVF, MRF, SH-SOB21, SH-SSD5, SH-ZNCC21, SUPER-rSGM5), (j) choice map. On this particular stereo pair, the processed choice map enables for an absolute reduction of the error rate of 7.41% Out-All and 7.27% Out-Noc with respect to the most accurate matcher, SUPER-rSGM5.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(11):2274–2282, nov 2012. 5

[2] Z. Chen, X. Sun, L. Wang, Y. Yu, and C. Huang. A deep visual correspondence embedding model for stereo matching

costs. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 972–980, 2015. 4

[3] M. Christian, R. Markus, F. Friedrich, and B. Horst. Using self-contradiction to learn confidence measures in stereo vision. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 4

[4] J. M. Cruz, G. Pajares, J. Aranda, and J. L. F. Vindel. Stereo matching technique based on the perceptron criterion function. *Pattern Recogn. Lett.*, 16(9):933–944, sep 1995. 2

[5] L. De-Maeztu, S. Mattoccia, A. Villanueva, and R. Cabeza. Linear stereo matching. In *A13th International Conference on Computer Vision (ICCV2011)*, November 6-13 2011. 2, 5

[6] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The kitti dataset. *Int. J. Rob. Res.*, 32(11):1231–1237, sep 2013. 1, 2, 4, 5

[7] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 1, 2, 4, 5

[8] A. Geiger, M. Roser, and R. Urtasun. Efficient large-scale stereo matching. In *Asian Conference on Computer Vision (ACCV)*, 2010. 5

[9] S. Ghosh, E. Laksana, S. Scherer, and L.-P. Morency. A Multi-label Convolutional Neural Network Approach to Cross-Domain Action Unit Detection. In *Proceedings of ACII 2015*, Xi'an, China, sep 2015. IEEE. 3, 5

[10] R. Haeusler, R. Nair, and D. Kondermann. Ensemble learning for confidence measures in stereo vision. In *CVPR. Proceedings*, pages 305–312, 2013. 1. 2

[11] K. He, J. Sun, and X. Tang. Guided image filtering. In *Proceedings of the 11th European Conference on Computer Vision: Part I*, ECCV'10, pages 1–14, Berlin, Heidelberg, 2010. Springer-Verlag. 5

[12] H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 30(2):328–341, feb 2008. 1, 2, 5

[13] A. Hosni, M. Bleyer, and M. Gelautz. Secrets of adaptive support weight techniques for local stereo matching. *Computer Vision and Image Understanding*, 117(6):620–632, jun 2013. 2

[14] A. Hosni, C. Rhemann, M. Bleyer, C. Rother, and M. Gelautz. Fast cost-volume filtering for visual correspondence and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 35(2):504 – 511, 2013. 2, 5

[15] X. Hu and P. Mordohai. A quantitative evaluation of confidence measures for stereo vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, pages 2121–2133, 2012. 2

[16] N. Komodakis, G. Tziritas, and N. Paragios. Fast, approximately optimal solutions for single and dynamic mrfs. In *2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), 18-23 June 2007, Minneapolis, Minnesota, USA*, 2007. 5

[17] D. Kong and H. Tao. A method for learning matching errors in stereo computation. In *British Machine Vision Conference (BMVC)*, 2004 2004. 2

[18] D. Kong and H. Tao. Stereo matching via learning multiple experts behaviors. In M. J. Chantler, R. B. Fisher, and E. Trucco, editors, *British Machine Vision Conference, BMVC*, pages 97–106. British Machine Vision Association, 2006. 2

[19] G. Kurata, B. Xiang, and B. Zhou. Improved neural network-based multi-label classification with better initialization leveraging label co-occurrence. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 521–526, June 2016. 3, 5

[20] L. Ladicky, P. Sturgess, C. Russell, S. Sengupta, Y. Bastanlar, W. F. Clocksin, and P. H. S. Torr. Joint optimization for object class segmentation and dense stereo reconstruction. *International Journal of Computer Vision*, 100(2):122–133, 2012. 2

[21] M. S. Lew, T. S. Huang, and K. Wong. Learning and feature selection in stereo matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(9):869–881, Sep 1994. 2

[22] O. Mac Aodha, A. Humayun, M. Pollefeys, and G. J. Brostow. Learning a confidence measure for optical flow. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(5):1107–1120, may 2013. 2

[23] N. Mayer, E. Ilg, P. Häusser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 3, 4

[24] M. Menze and A. Geiger. Object scene flow for autonomous vehicles. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 1, 2, 4

[25] A. Motten, L. Claesen, and Y. Pan. *Trinocular Stereo Vision Using a Multi Level Hierarchical Classification Structure*. 2013. 1, 2

[26] M.-G. Park and K.-J. Yoon. Leveraging stereo matching with learning-based confidence measures. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 2

[27] M. Poggi and S. Mattoccia. Learning a general-purpose confidence measure based on o(1) features and asmarter aggregation strategy for semi global matching. In *Proceedings of the 4th International Conference on 3D Vision, 3DV*, 2016. 2

[28] N. Sabater, A. Almansa, and J.-M. Morel. Meaningful Matches in Stereovision. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 34(5):930–42, dec 2011. 2

[29] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. Comput. Vision*, 47(1-3):7–42, apr 2002. 1, 2

[30] D. Scharstein and R. Szeliski. High-accuracy stereo depth maps using structured light. In *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, CVPR'03, pages 195–202, Washington, DC, USA, 2003. IEEE Computer Society. 1, 2, 4

[31] R. Spangenberg, T. Langner, S. Adfeldt, and R. Rojas. Large scale semi-global matching on the cpu. In *IV*, 2014. 5

[32] A. Spyropoulos, N. Komodakis, and P. Mordohai. Learning to detect ground control points for improving the accuracy of stereo matching. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1621–1628. IEEE, 2014. 2

[33] A. Spyropoulos and P. Mordohai. Ensemble classifier for combining stereo matching algorithms. In *Proceedings of the 2015 International Conference on 3D Vision*, 3DV '15, pages 73–81, 2015. 1, 2, 4, 5, 6, 7

[34] J. Sun, N.-N. Zheng, and H.-Y. Shum. Stereo matching using belief propagation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(7):787–800, jul 2003. 2

[35] E. Tola, V. Lepetit, and P. Fua. Daisy: An efficient dense descriptor applied to wide-baseline stereo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(5):815–830, may 2010. 5

[36] F. Tombari, S. Mattoccia, L. D. Stefano, and E. Addimanda. Classification and evaluation of cost aggregation methods for stereo correspondence. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, June 2008. 2

[37] Y. Wei, W. Xia, J. Huang, B. Ni, J. Dong, Y. Zhao, and S. Yan. CNN: single-label to multi-label. *CoRR*, 2014. 3, 5

[38] K. Yamaguchi, T. Hazan, D. McAllester, and R. Urtasun. Continuous markov random fields for robust stereo estimation. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part V*, ECCV'12, pages 45–58, 2012. 2

[39] S. Zagoruyko and N. Komodakis. Learning to compare image patches via convolutional neural networks. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[40] J. Zbontar and Y. LeCun. Computing the stereo matching cost with a convolutional neural network. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2, 4

[41] J. Zbontar and Y. LeCun. Stereo matching by training a convolutional neural network to compare image patches. *Journal of Machine Learning Research*, 17:1–32, 2016. 1, 2, 4, 5

[42] L. Zhang and S. M. Seitz. Estimating optimal parameters for mrf stereo from a single image pair. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(2):331–342, 2007. 2

[43] S. Zhu, L. Zhang, and H. Jin. A locally linear regression model for boundary preserving regularization in stereo matching. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part V*, ECCV'12, pages 101–115, 2012. 2