2016 Fourth International Conference on 3D Vision

Learning a general-purpose confidence measure based on $O(1)$ features and a smarter aggregation strategy for semi global matching

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

Inferring dense depth from stereo is crucial for several computer vision applications and Semi Global Matching (SGM) is often the preferred choice due to its good trade-off between accuracy and computation requirements. Nevertheless, it suffers of two major issues: streaking artifacts caused by the Scanline Optimization (SO) approach, at the core of this algorithm, may lead to inaccurate results and the high memory footprint that may become prohibitive with high resolution images or devices with constrained resources. In this paper, we propose a smart scanline aggregation approach for SGM aimed at dealing with both issues. In particular, the contribution of this paper is threefold: i) leveraging on machine learning, proposes a novel general-purpose confidence measure suited for any for stereo algorithm, based on $O(1)$ features, that outperforms state-of-the-art ii) taking advantage of this confidence measure proposes a smart aggregation strategy for SGM enabling significant improvements with a very small overhead iii) the overall strategy drastically reduces the memory footprint of SGM and, at the same time, improves its effectiveness and execution time. We provide extensive experimental results, including a cross-validation with multiple datasets (KITTI 2012, KITTI 2015 and Middlebury 2014).

1. Introduction

Stereo is a well-known technique to infer dense depth data from two or more images and several approaches have been proposed to deal with this problem. However, accuracy in challenging conditions still remains an open research issue. This fact has been clearly emphasized with recent datasets [9, 20, 25] made of scenes acquired in realistic and difficult environments. While some pitfalls are intrinsically related to the stereo setup, such as occlusions [6] and distinctiveness [19], others, such as low signal-to-noise ratio and untextured regions, typically occur in challenging environmental conditions characterized by poor illumination, reflective surfaces and so on. In practical applications, it is also important to effectively detect unreliable depth measurements by means of a point-wise *confidence measure* that encodes the degree of uncertainty. Confidence measures, extensively reviewed and evaluated in [16], are computed according to different strategies based on analysis of cost curves. Recently, some authors [10, 27, 21] showed how machine learning frameworks can improve the reliability of confidence measures. A common trend in these works is the joint use of a pool of confidence measures to define a feature vector, fed to a classifier (e.g., random forest (RF)), that allows to improve their effectiveness with respect to each of the considered individual measures.

Concerning stereo algorithms, the Semi Global Matching (SGM) algorithm [12] has become very popular due to its good trade-off between accuracy and computation requirements. For this reason it has been implemented, according to different strategies and simplifications, on almost any computing architecture. SGM relies on multiple disparity optimization steps performed along different paths, typically 8 or 16. Disparity optimization is performed, by means of the Scanline Optimization (SO) [24] algorithm, minimizing an energy function. Although SO is very fast, disparity optimization on a 1D domain may lead to well-known *streaking* artifacts. SGM partially attenuates this problem by aggregating energy computed along different paths and by selecting, by means of a winner-takes-all (WTA) strategy, the disparity label with the minimum cost.

In this paper we take a deeper look at SGM with the aim to improve its accuracy by softening the propagation of streaking artifacts induced by SO. For this purpose, we

978-1-5090-5407-7/16 $31.00 © 2016 IEEE
DOI 10.1109/3DV.2016.61

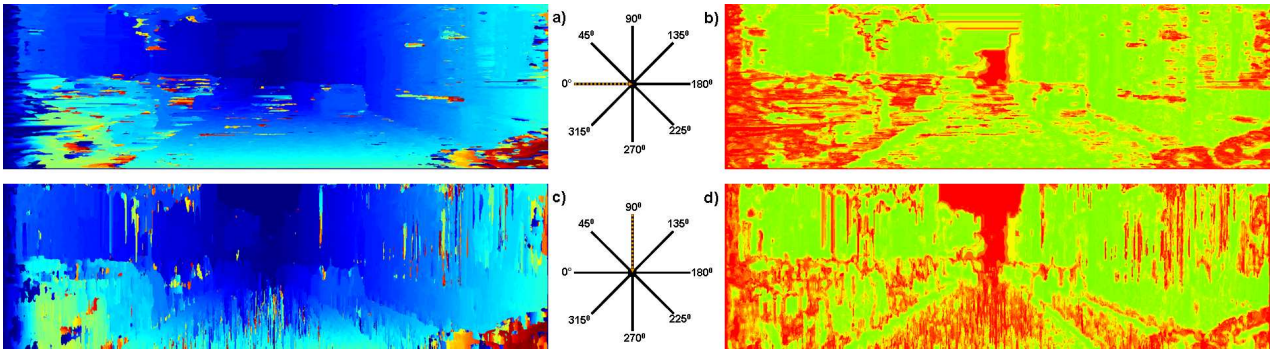

Figure 1. Streaking detection step. (a) Disparity map obtained from path 0°, (b) streaking detection on path 0°, (c) disparity map obtained from path 90°, (d) streaking detection on path 90°(c). Disparity maps a) and (c) are encoded with warmer colors for closer points and colder colors for farther points. Confidence maps b) and d) depicts unreliable and reliable assignments, respectively, in red and green.

propose a framework based on an ensemble classifier (in particular, RF) that allows us to obtain a very effective and general-purpose confidence measure by processing features extracted from a disparity map. Then, we apply our confidence measure to the output of each single SO of SGM in order to detect streaking and thus softening its effect when aggregating costs from different paths.

In particular, focusing on SGM, we extract from the disparity map obtained along each path, with a WTA strategy, a pool of $O(1)$ features processed by our framework to obtain a confidence measure that encodes the degree of uncertainty of each SO. The outcome of this analysis is then fed to a smart aggregation step that, conversely from SGM, weights each path according to the estimated uncertainty in order to obtain a more accurate overall disparity map. We thoroughly evaluate effectiveness of our general-purpose confidence measure as well as the disparity accuracy achieved by our overall proposal, referred to as $RF - SGM$, on KITTI 2012 [9]. Moreover, to avoid *overfitting* and to prove that it can generalize its behavior to different scenes, we cross-validated our method on KITTI 2015 [20] and Middlebury 2014 [25] performing training on eight stereo stereo pairs from the KITTI 2012 dataset. In both evaluations, experimental results confirm that our proposal increases the accuracy of the original SGM algorithm with a minimal overhead and, by adopting appropriate strategy discussed later, enables obtaining better results with a reduced execution time and at a fraction of the original memory footprint. Moreover, to validate our $O(1)$ feature set, we compare the performance of our proposal when fed with such features and with the features proposed in [21]. This evaluation shows that our general-purpose confidence measure outperforms state-of-the-art.

2. Related work

According to [24], stereo matching algorithms can be categorized into two broad categories, *local* and *global* methods. Both perform a subset of the following four steps: 1) matching cost computation 2) cost aggregation 3) disparity computation/optimization 4) disparity refinement. Local methods [29, 14, 15, 5] typically focus on steps 1 and 2 while global methods [18] mostly on 1 and 3. Although local method can be very fast, according to recent evaluations on challenging datasets [9, 20, 25] they are clearly outperformed by global methods. Among these latter approaches, a good trade-off between accuracy and execution time is represented by the semi global method proposed by Hirschmuller [12]. This strategy, described in details in the next section, independently enforces on multiple paths, by means of the SO algorithm [24], a *smoothness* constraint and sums up the outcome of each one. The optimal disparity is assigned according to a WTA strategy applied to the final aggregated costs. This method, according to [9, 20, 25], is adopted by most top-performing algorithms such as [30, 31] and [3]. Moreover, original or variants of SGM have been implemented on different computing architectures such as GPUs [30, 31], FPGAs [1, 8] and other embedded devices. A review and evaluation [2] of SGM variants based on modulation of the smoothing penalty according to image content highlights that, for structured environments, constant penalty terms are appropriate. Spangenberg et al. proposed *weighted* SGM aimed at weighting the costs of each each path according to its compliance with the associated surface normal [26]. A memory efficient, yet simplified, version of SGM, referred to as eSGM, yielding almost equivalent error rate with respect to the original algorithm has been proposed in [13]. Finally, the MGM [7] algorithm aims at improving the accuracy of SGM according to a *more global* strategy.

Concerning the cost computation step, extensively reviewed and evaluated in [11], *non-parametric* approaches, such as the *census* transform, are often a preferred choice due to their effectiveness [11]. More recently, some authors proposed cost computation learned by means of Convolutional Neural Networks (CNN). This strategy, turned out to

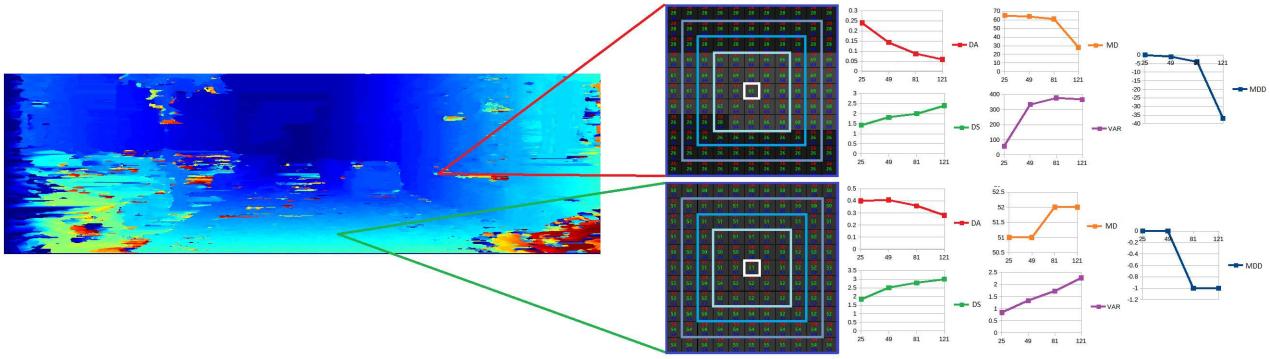

Figure 2. Overview of the proposed streaking detection approach for two points of a disparity map obtained along path $0°$ by SO. For each point, features are extracted from patches of increasing size (encoded with different shades of blue). The behaviors of the features extracted on patches from size 5×5 (25) to size 11×11 (121) is reported on the right for the two points: one (top) belonging to an area with streaking and one (bottom) within a region without streaking.

be very effective as reported in [30, 31] and [3].

Strictly related to stereo matching are confidence measures, reviewed and evaluated by Hu and Mordohai [16], aimed at detecting unreliable disparity assignments by analyzing the cost curve. It has been shown that, combining multiple confidence measure by means of machine learning approaches, yields to significant improvements [10, 27, 21]. In particular, the confidence measure proposed by Park et al. [21], according to the methodology proposed in [16], outperforms state-of-the-art. Reliable confidence measures can be used to improve disparity accuracy by selecting reliable *ground control points* [27] or by modulating matching costs as proposed in [21]. Finally, machine learning has been deployed also to combine multiple stereo algorithms with a RF [28] and with a CNN [23].

3. Semi Global Matching

Semi Global Matching [12] represents a good trade-off between accuracy and computational complexity and for this reason it is very popular. Most of the top-performing algorithms in the literature rely on such method to obtain state-of-art results according to standard evaluation datasets [9, 20, 25]. For each pixel p, SGM combines the outcome of multiple energy minimizations computed by independent SO [24] algorithm on different paths $s \in S$, typically 8 or 16 according to [12]. For the 8 path version, referred to as SGM_8, the paths $S = \{0°, 45°, 90°, 135°, 180°, 225°, 270°, 315°\}$ are depicted in Figure 1. Each SO, within the disparity range $[0, d_{max}]$ and along each path $s \in S$, performs for each pixel p a disparity optimization according to the following energy term $E_s(p, d)$,

$$E_s(p,d) = C(p,d) + \min\Big\{E_s(p',d), E_s(p',d-1)+P1, E_s(p',d+1)+P1, \min_{i\in[0,d_{max}]}(E_s(p',i)+P2)\Big\} - \min_{i\in[0,d_{max}]}(E_s(p',i)) \quad (1)$$

where p' represents the previous pixel along the path and $C(p, d)$ the *point-wise* or aggregated matching cost computed, for each disparity $d \in [0, d_{max}]$, between reference and target corresponding points along epipolar lines. Terms $P1$ and $P2$ ($P1 < P2$) in (1) enforce *smoothing* by penalizing disparity variations along each path s. According to [11], among the many cost functions proposed for stereo *non-parametric* approaches such as *census* perform very well in challenging environments. Compared to global approaches that enforce a smoothness term on a grid (i.e., 2D domain) SO is less computational demanding. However, it is well-know that it is prone to streaking artifacts along the direction of the path. SGM softens this effect by summing up, for each point p, the results yielded by multiple SO as follows

$$E(p,d) = \sum_{s\in S} E_s(p,d) \quad (2)$$

and a selects the optimal disparity assignment according to a WTA strategy. The SGM algorithm requires to store the entire Disparity Space Image (DSI) [24] resulting in a high memory footprint. Moreover, strategy (2) attenuates streaking artifacts only partially. Our proposal aims at tackling both issues by learning a smarter aggregation strategy with respect to (2) driven by an analysis of the outcome of the SOs computed along each path $s \in S$.

4. Proposed method

In this paper, we introduce a novel step within the stereo pipeline of the SGM algorithm to detect streaking artifacts occurring on each path with the aim to soften their propagation in the final disparity map. Streaking detection for each SO is carried out by means of a RF-based framework and then used to weight, accordingly, the contribution brought in by each scanline.

In this section, we introduce the feature vector adopted for our streaking detection module and we discuss the importance of the variables obtained through the training process. Then, we introduce a smart scanline aggregation approach that takes into account such confidence values to refine the final DSI.

4.1. Features extraction

We process a feature vector, through a RF framework, in order to infer, for each pixel p and path $s \in S$, a value $C_s(p)$ that encodes its degree of reliability ($\in [0, 1]$). Five cues are computed on four patches of increasing size $\Omega = \{5\times5, 7\times7, 9\times9, 11\times11\}$ centered on p. By observing the behavior of the streaking effect, which typically occurs near depth discontinuities, we extract features that enable to encode the statistical dispersion of disparity in the neighborhood of p. We define H_p^N the histogram of disparity within patch $N \in \Omega$ centered in p, $H_p^N(d)$ the amount of points at disparity d within N, and the cardinality N as:

$$\bar{\bar{N}} = \sum_{d\in[0,d_{max}]} H_p^N(d) \tag{3}$$

Given a patch N, centered on p, we extract the following cues from the disparity map:

1. Disparity agreement (DA), encodes the number of neighboring pixels with the same disparity of the central point p:

$$DA_p^N = H_p^N(d(p)) \tag{4}$$

   A large amount of pixels sharing the same disparity of p stands for a higher likelihood of correctness with respect to circumstances where p has slighter support from its neighbors.

2. Disparity scattering (DS), encodes how many different disparity hypotheses appears in the neighborhood of p

$$DS_p^N = -log\frac{\sum_{d\in[0,d_{max}]} 1 - \delta(H_p^N(d), 0)}{\bar{\bar{N}}} \tag{5}$$

   where δ is Kronecker's delta function (1 if $H_p^N(d)$ value is zero, 0 otherwise). According to such definition, a patch of $\bar{\bar{N}}$ pixels in complete disagreement with $d(p)$ yields to a DS value equal to zero. The lower the number of different hypotheses within the patch, the higher the value of the DS score is.

3. Median disparity (MD)

$$MD_p^N = median(H_p^N) \tag{6}$$

4. Variance of the disparity values (VAR),

$$VAR = \frac{1}{\bar{\bar{N}}} \sum_{q\in N} (d(q) - \mu(p))^2 \tag{7}$$

   with

$$\mu(p) = \frac{1}{\bar{\bar{N}}} \sum_{q\in N} d(q) \tag{8}$$

5. Median deviation of disparity (MDD), as proposed in [27, 21], which is the negative of the absolute difference between the disparity in p and the median disparity value in the patch N,

$$MDD = - \mid d(p) - MD(H_p^N) \mid \tag{9}$$

For each disparity map estimated by SO on path p, we combine these 5 features at four scales $N \in \Omega$ obtaining the following features vector, $f_{20} = [f_1, f_2, \ldots, f_{19}, f_{20}]^T$.

By leveraging on a multi-scale approach, more information is provided to the RF to identify potentially erroneous matches. In particular, in presence of a streaking, with larger patches the magnitude of the features encoding the statistical dispersion decreases. Figure 2 gives an overview of the multi-scale approach described, emphasizing the different behavior of each feature and for each patch size in two completely different circumstances (with streaking, on top, and without streaking). It worth to note that the proposed features can be computed in constant time exploiting $O(1)$ techniques such as *integral images* for VAR and histogram-based optimization techniques [22, 4, 17] for DA, DS, MD and MDD. Compared to features extracted analyzing the behavior of the cost curve [27, 21] with complexity $O(d_{max})$, all our features are independent of the disparity range as well as of the patch size and hence turn out to be $O(1)$.

We train an ensemble regression trees classifier that provides a confidence value $C_s(p)$ for each path. It is worth observing that, according to the proposed strategy, we can specialize the RF for each path s or we can train the RF

on multiple paths obtaining a more general RF suited for any path. We'll provide a detailed discussion of two strategies, respectively referred to as *multiple* (M) and *single*, in the experimental results section. Moreover, we point out that the computation of the disparity map for each $s \in S$ required by our approach introduces a negligible overhead being, substantially, the outcome of SO.

Finally, differently from [21], we do not consider false positives, false negatives, true positive and true negatives to rescale our confidence, in order to maintain the gap between lower and higher values. In fact, during the experimental evaluation we tested either raw and rescaled values, obtaining no substantial difference between the two approaches. Moreover, the former strategy allows us to enhance more effectively the costs of reliable scanlines.

4.2. Smart aggregation

Given a point p, the smart aggregation approach proposed aims at replacing the cost aggregation performed by SGM on each path computed by the SO algorithm with a strategy that takes into account the reliability $C_s(p)$ of each path $s \in S$ estimated by the RF. Specifically, for each point p, we aggregate the SO costs according to the following weighted sum:

$$E^*(p,d) = \frac{\sum_{s \in S} C_s(p) E_s(p,d)}{\frac{1}{S} \sum_{s \in S} C_s(p)} \tag{10}$$

in (10) the average confidence value at the denominator allows us to further enhance the dynamic of the cost curve whenever a path s is expected to be more reliable with respect to the others. Although never occurred in our experimental evaluation, if all the $C_s(p)$ are zero we replace $E^*(p,d)$ with $E(p,d)$ and hence assign the disparity according to the conventional SGM approach.

In the next section we prove that the learned aggregation strategy outlined so far enables to improve the effectiveness of the SGM algorithm. Moreover, with a subset of appropriate paths $s \in S$, we are able to obtain better results with respect to the standard SGM approach. This strategy also enables us to reduce the execution time and the memory footprint of SGM making our proposal suited to higher resolution stereo pairs and computing architectures with constrained resources.

5. Experimental results

In this section, we provide an exhaustive evaluation of our proposal on standard dataset KITTI 2012 detailing the methodology adopted to train the RF in two different configurations (i.e., single and multiple RF) and evaluating, by means of Area Under the Curve (AUC) [16], the confidence measure $C_s(p)$ provided by our framework. Then, we report on the same dataset, the improvements yield by our framework with respect to SGM_8. Moreover, to prove that our framework is able to generalize to different scenarios, we provide additional experimental results, in terms of AUC and improvements with respect to SGM_8, regarding the cross-validation on KITTI 2015 and Middlebury 2014 with the RF trained on KITTI 2012. Finally, we show that our method outperforms state-of-the-art [21] and also report an experimental evaluation combining the two approaches.

5.1. Framework configuration and training

In our experiments, we adopt as baseline the SGM_8 [12] algorithm computed on the 8 paths belonging to S depicted in Figure 1. As matching cost function we use the Hamming distance, aggregated on 5×5 patches, computed on images obtained according to a binary census transform considering 5×5 neighborhood points. We set parameters $P1$ and $P2$ of the SGM algorithm to 30 and 300, respectively. According to [2] we do not change these parameters being the target datasets quite structured.

We tuned an ensemble classifier made of 10 regression trees, maximum depth equals to 25 and minimum number of samples in each node to split equal to 20. To generate the training data, we processed eight challenging stereo pairs from KITTI 2012 commonly adopted on related works [10, 21], which are 43, 71, 82, 87, 94, 120, 122, and 180^{th}. For each of these stereo pairs, eight independent SOs provide a disparity map for each path according to the WTA strategy. We evaluate the performance of our proposal with a single RF as well as with one RF for each path. It is worth observing that, in this latter case, the amount of training sample on the same images is reduced by a factor 8. Moreover, we trained two versions of our framework: one with our features and the other with the features proposed in [21] in order to provide a comparison with state-of-art. The evaluation was carried out on the remaining images of the KITTI 2012 dataset and also cross-validated (with the same training) on KITTI 2012, KITTI 2015 and Middlebury 2014 datasets. For the latter case we used images at quarter resolution.

5.2. Confidence evaluation

In this section we evaluate the confidence provided by the proposed RF framework to compare its effectiveness with respect to state-of-art approaches. For this purpose we compute AUC, a common method [16, 27, 10, 21] to evaluate the effectiveness of a confidence measure. Given a confidence map, pixels are sorted according to their confidence in descending order. A subset of them equal to 5% of the total is sampled and the error rate is plotted, then the subset is increased to 10% of the total and so on until 100%. Ties are solved by taking into the subset all the points with the same confidence value. Given the percentage of erro-

Dataset	*Optimal*	PKRN	LRD	Park	Proposed	Park (M)	Proposed (M)
KITTI 2012	0.038202	0.182604	0.155302	0.075122	0.072264	0.092018	**0.071020**
KITTI 2015	0.043930	0.193883	0.160993	0.098198	0.092410	0.111211	**0.089296**
Middlebury 2014	0.050107	0.181431	0.172787	0.093343	**0.084257**	0.112117	0.084539

Table 1. Evaluation of the confidence, in terms of average AUC, provided by our framework, in single and M configuration, using our features and those proposed in [21], compared with optimal values and confidence measures from literature [16]. Top table, results for KITTI 2012 . Central table, results for KITTI 2015. Bottom table, results for Middlebury 2014.

neous points ϵ, according to [16], the optimal AUC can be obtained as $\epsilon + (1-\epsilon)ln(1-\epsilon)$. AUC closer to the optimal value reflects a better confidence prediction.

Table 1 reports tables containing average AUCs computed on KITTI 2012 (first row), KITTI 2015 (second row) and Middlebury 2014 (third row) datasets, evaluating it over the results of each SO and averaging. The tables report optimal values and AUCs related to PKRN [16], LRD [16], Park et al. [21], our proposal, Park et al. trained on configuration M and our proposal trained on configuration M.

The numbers show that our feature vector f_{20} significantly outperforms in most cases state-of-the-art [21]. Moreover, we can notice that configuration M yields even more accurate results in terms of confidence prediction. On average, on all the eight paths, the relative improvement in terms of AUC with respect to [21] adopting our features is 22% on KITTI 2012, 20.4% on KITTI 2015 and 24.6% on Middlebury 2014 for configuration M and, respectively, 3%, 6.6% and 9.7% with a single RF. Regarding our proposal, the relative improvement yielded by configuration M with respect to training a single RF is 1.7% on KITTI 2012, 3.3% on KITTI 2015 and -0.3% on Middlebury 2014.

Summarizing, configuration M clearly performs better when compared to [21] with an average relative improvement of 22.3%. On the other hand, when comparing our method in the two configurations proposed, on the Middlebury dataset we do not have a dominant strategy for all the eight scanlines.

To further confirm the effectiveness of the proposed features, regardless to its application to the smarter aggregation strategy described so far, we evaluated AUC values for confidence measures yielded by our proposal and [21] with two different algorithms: Block Matching (BM) and the outcome of the full SGM_8 method (*i.e.*, not the single SOs). Table 2 reports average results on KITTI 2015 and Middlebury 2014, showing how our general-purpose confidence measure clearly outperforms [21] even considering the output of generic stereo algorithms outside the smarter aggregation context previously proposed.

5.3. Disparity accuracy evaluation

In this section, we assess the performance of our proposal, referred to as $RF - SGM_8$, by gathering the absolute improvement in terms of error rate, with respect to the baseline SGM_8 algorithm, with our features and with

Algo	Optimal	Park et al. [21]	Proposed	Win rate
BM	0.137	0.179	**0.163**	200/200
SGM_8	0.038	0.124	**0.095**	197/200
BM	0.093	0.114	**0.106**	13/15
SGM_8	0.042	0.093	**0.063**	15/15

Table 2. Confidence measures evaluation, in terms of average AUC, provided by our method and [21] on two popular stereo algorithms: Block Matching (BM) and the standard SGM_8 algorithm [12]. On top ad bottom, respectively, results on KITTI 2015 and Middlebury 2014 dataset. Our confidence measure constantly outperforms state-of-the art method [21] with both algorithms, confirming that its effectiveness is not restricted to SO.

the features proposed in [21]. Moreover, we include in this evaluation the results gathered by our own implementation of the DSI modulation proposed in [21].

Figures 3 and 4-left report the absolute disparity accuracy improvement on KITTI 2012, KITTI 2015 and Middlebury 2014 obtained by $RF - SGM_8$, with our features in both configurations, with respect to baseline SGM_8. On KITTI datasets configuration M outperforms the single RF. In particular, on average, SGM_8 achieves a 9.90% error rate on KITTI 2012 and 9.56% on KITTI 2015. $RF - SGM_8$ in single RF configuration achieves, respectively, 9.38% and 9.14% (-0.52% and -0.42%) while configuration M achieves 9.26% and 9.04% (-0.64% and -0.52%). Conversely, on average, on the Middlebury dataset the single RF performs slightly better than configuration M. In fact, SGM_8 has an error rate of 22.93%, single RF 21.50% (-1.43%) and configuration M 21.60% (-1.33). These accuracy improvements follow the behavior of the confidence measure C_s analyzed in the previous section.

Figures 5 and 4-right report the absolute accuracy improvement yielded by our framework, in configuration M, for KITTI datasets (Figure 5) and single RF for Middlebury dataset (Figure 4-right), using our framework with the proposed features and with those of Park et al. On the three datasets, our feature vector is always more effective than state-of-the-art when deployed with the smart aggregation strategy proposed. In particular, on average, with the feature vector [21] we obtain 9.40% ($+0.14$) on KITTI 2012, 9.14% ($+0.10$) on KITTI 2015 and 22.47% ($+0.97$) on Middlebury.

It is worth to point out that, extending the training set by a factor 8 slightly improves the performance of config-

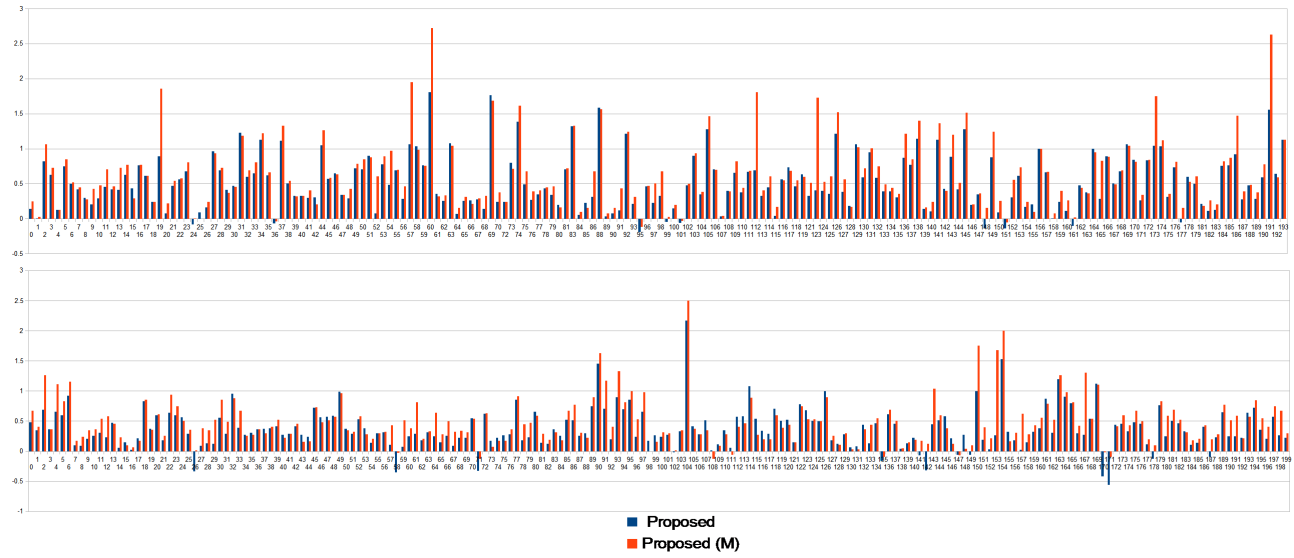

Figure 3. Absolute improvement of disparity accuracy yielded by our approach with respect to SGM_8 on KITTI 2012 (top) and KITTI 2015 (bottom). The improvements introduced by a single RF (blue) and by configuration M (red).

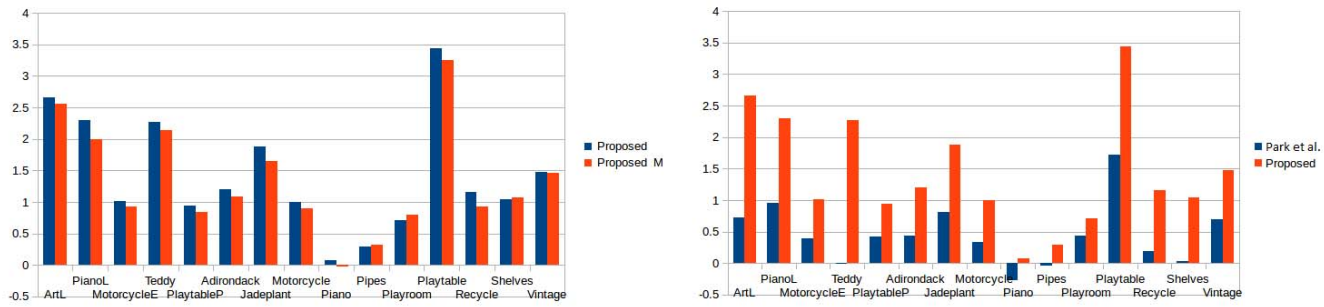

Figure 4. (Left) Absolute improvement of disparity accuracy yielded by our approach with respect to SGM_8 on Middlebury 2014: single RF (blue) and configuration M (red). (Right) Absolute improvement of disparity accuracy with respect to SGM_8 adopting our features (red) and features [21] (blue) on Middlebury 2014 with configuration M.

uration M on the Middlebury dataset but does not allow in the tested cases to outperform the single RF. Nevertheless, regarding the comparison of feature vector, on the three datasets our proposal outperforms [21] in any configuration and amount of training samples.

We also compared our proposal with the DSI modulation proposed in [21], referred to as $PARK_8$, applied to our baseline SGM_8 algorithm. According to this evaluation we obtained, on average, with $RF-SGM_8$ an absolute improvement with respect to $PARK_8$ of 1.04% on KITTI 2012 and of 0.69% on KITTI 2015. Finally, we evaluated the combination of $PARK_8$ and the proposed $RF-SGM_8$ obtaining an absolute improvement with respect to SGM_8 of 1.04 on KITTI 2012 and of 1.14 on KITTI 2015.

For each path, on KITTI datasets: without specific optimizations, our C++ implementation of SO requires $4s$, computing feature vector $0.08s$ and confidence measure computation $0.53s$. The final smart aggregation phase introduces a negligible overhead. Therefore, our method substantially adds an overhead of 15% to the overall execution time. Computing our feature vector is $O(1)$ and the memory footprint of the RF framework is independent of image resolution and, only for configuration M, proportional to the number of paths. The high memory footprint is a major issue of the SGM algorithm, particularly relevant with computing architectures with constrained memory resources. However, this fact may be critical with any device when dealing with high resolution stereo pairs. For instance, the full resolution Middlebury dataset has images of size $W \times H = 3000 \times 2000$ with a disparity range d_{max} = 800. In this very case, the footprint of the only DSI would be $\propto$ to 9 GB, using 16 bit *short* for aggregated costs. This amount of memory might be prohibitive with any current computing device including standard PCs. On

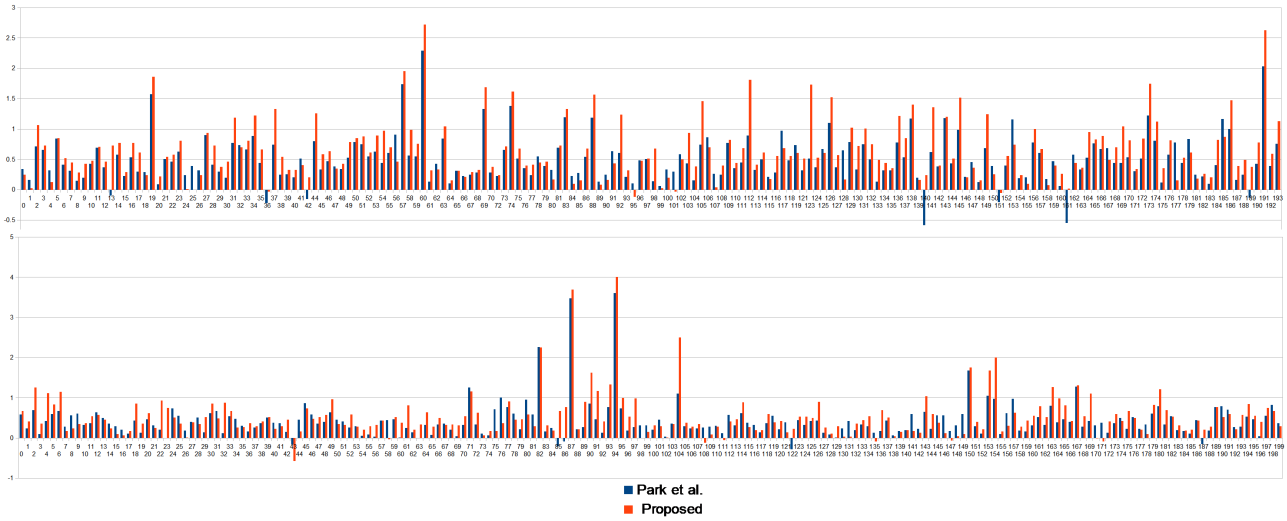

Figure 5. Absolute improvement of disparity accuracy with respect to SGM_8 adopting our features (red) and features [21] (blue) on KITTI 2012 (top) and KITTI 2015 (bottom). In both cases, results are concerned with configuration M.

the other hand, it is worth observing that using a subset of S made of paths $0°, 45°, 90°$ and $135°$ the SGM algorithm, referred to as SGM_4, would have a memory footprint reduced by a factor $(H+3)/3$. For the full resolution Middlebury dataset this factor is about 667 (memory footprint of SGM_4 about 13.8 MB), for KITTI datasets is about 124 (memory footprint of SGM_4 about 1.9 MB). Even compared to the memory-efficient eSGM [13] (providing results almost equivalent to the vanilla SGM_8 adding, however, a further image scan), our approach enables a notable reduction of the memory footprint improving at the same time the overall accuracy. The memory of $RF-SGM_4$ is reduced, with respect to eSGM, by a factor almost 10 on the KITTI dataset and by a factor almost 16 on the full-resolution Middlebury 2014 dataset. Moreover, with the huge resolution stereo pairs reported in the eSGM paper [13], the memory footprint of $RF-SGM_4$ is 0.03 GB, 4.8 GB for eSGM and 272 GB for SGM_8 .

Although the SGM_4 does not provide the same accuracy of SGM_8 (and eSGM), it has been widely adopted, at the expense of reduced performance with respect to SGM_8, when the memory footprint represents the major constraint [1, 8]. Nevertheless, on the same four paths previously highlighted, the proposed method, referred to as $RF-SGM_4$, clearly outperforms SGM_8 as reported in Table 3 on KITTI 2012, KITTI 2015 and Middlebury. This interesting fact can be exploited to reduce the execution time of SGM_8 and, more importantly, to drastically reduce the memory footprint without compromising its overall effectiveness in order to fit with a broader class of devices and image resolutions. Observing the table we can notice that, on average, on the three datasets $RF-SGM_4$ improves the disparity accuracy with respect to SGM_8 of 0.44% deploying only 4 paths and hence enabling a drastically reduced memory footprint.

Dataset	K12	K15	M14	avg.
SGM_8	9.90%	9.59%	22.92%	**14.13%**
$RF-SGM_8$	9.26%	9.04%	21.49%	13.26%
SGM_4	10.65%	11.19%	23.50%	15.11%
$RF-SGM_4$	9.41%	9.60%	22.07%	**13.69%**

Table 3. Average error achieved by the SGM algorithm on 8 SGM_8 and 4 SGM_4 paths and by our RF-SGM approach on 8 $RF-SGM_8$ and 4 $RF-SGM_4$ paths.

6. Conclusions

In this paper, leveraging on machine learning, we have: i) proposed a novel general-purpose confidence measure for stereo matching based on O(1) features uniquely computed in the disparity domain ii) focusing our attention on the popular and effective SGM algorithm, we have exploited our confidence measure to propose a smarter aggregation framework aimed at increasing the effectiveness of SGM with a negligible overhead c) the overall framework allows us to achieve, with respect to SGM, comparable or better accuracy with a notably lower memory footprint thus dealing with one of the major issues of this algorithm. Exhaustive experimental results on KITTI 2012, KITTI 2015 and Middlebury 2014 confirmed the effectiveness of our proposals.

Acknowledgments The authors would like to acknowledge with thanks Valentina Cremonini, Simone Nigro and Matilde Ugolini for their contribution to the implementation of this work.

References

[1] C. Banz, S. Hesselbarth, H. Flatt, H. Blume, and P. Pirsch. Real-time stereo vision system using semi-global matching disparity estimation: Architecture and fpga-implementation. In *ICSAMOS*, pages 93–101, 2010. 2, 8

[2] C. Banz, P. Pirsch, and H. Blume. Evaluation of Penalty Functions for Semi-Global Matching Cost Aggregation. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pages 1–6, jul 2012. 2, 5

[3] 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. 2, 3

[4] D. Cline, K. White, and P. Egbert. Fast 8-bit median filtering based on separability. In *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, volume 5, pages V – 281–V – 284, Sept 2007. 4

[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

[6] G. Egnal and R. P. Wildes. Detecting binocular half-occlusions: Empirical comparisons of five approaches. *IEEE Transaction on Pattern Analysis and Machine Intelligence (PAMI)*, 24(8):1127–1133, 2002. 1

[7] G. Facciolo, C. de Franchis, and E. Meinhardt. Mgm: A significantly more global matching for stereovision. In *British Machine Vision Conference (BMVC)*, 2015. 2

[8] S. K. Gehrig, F. Eberli, and T. Meyer. A real-time low-power stereo vision engine using semi-global matching. In *ICVS*, pages 134–143, 2009. 2, 8

[9] 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, 3

[10] R. Haeusler, R. Nair, and D. Kondermann. Ensemble learning for confidence measures in stereo vision. In *CVPR. Proceedings*, pages 305–312, 2013. 1. 1, 3, 5

[11] H. Hirschmller. Evaluation of cost functions for stereo matching. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007. 2, 3

[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, 3, 5, 6

[13] H. Hirschmuller, M. Buder, and I. Ernst. Memory efficient semi-global matching. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume I-3, 2012*, pages 371–376, 2012. 2, 8

[14] 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

[15] 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

[16] 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. 1, 3, 5, 6

[17] M. Kass and J. Solomon. Smoothed local histogram filters. *ACM Trans. Graph.*, 29(4):100:1–100:10, jul 2010. 4

[18] 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. 2

[19] R. Manduchi and C. Tomasi. Distinctiveness maps for image matching. In *Image Analysis and Processing, 1999. Proceedings. International Conference on*, pages 26–31. IEEE, 1999. 1

[20] M. Menze and A. Geiger. Object scene flow for autonomous vehicles. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 1, 2, 3

[21] 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. 1, 2, 3, 4, 5, 6, 7, 8

[22] S. Perreault and P. Hbert. Median filtering in constant time. *IEEE Transactions on Image Processing*, 16(9):2389–2394, 2007. 4

[23] M. Poggi and S. Mattoccia. Deep stereo fusion: combining multiple disparity hypotheses with deep-learning. In *Proceedings of the 2016 International Conference on 3D Vision*, 2016. 3

[24] 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, 3

[25] 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, 3

[26] R. Spangenberg, T. Langner, and R. Rojas. Weighted semi-global matching and center-symmetric census transform for robust driver assistance. In *Computer Analysis of Images and Patterns - 15th International Conference, CAIP 2013, York, UK, August 27-29, 2013, Proceedings, Part II*, pages 34–41, 2013. 2

[27] 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. 1, 3, 4, 5

[28] 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. 3

[29] 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

[30] 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, 3

[31] 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. 2, 3