Computer Vision and Image Understanding
journal homepage: www.elsevier.com

Learning a confidence measure in the disparity domain from O(1) features

Matteo Poggi[a,], Fabio Tosi[a], Stefano Mattoccia[a]

[a]*University of Bologna, Viale del Risorgimento 2, 40136, Bologna, Italy.*

ABSTRACT

Depth sensing is of paramount importance for countless applications and stereo represents a popular, effective and cheap solution for this purpose. As highlighted by recent works concerned with stereo, uncertainty estimation can be a powerful cue to improve accuracy in stereo. Most confidence measures rely on features, mainly extracted from the cost volume, fed to a random forest or a convolutional neural network trained to estimate match uncertainty. In contrast, we propose a novel strategy for confidence estimation based on features computed in the disparity domain, making our proposal suited for any stereo system including COTS devices, and in constant time. We exhaustively assess the performance of our proposals, referred to as O1 and O2, on KITTI and Middlebury datasets with three popular and different stereo algorithms (CENSUS, MC-CNN and SGM), as well as a deep stereo network (PSM-Net). We also evaluate how well confidence measures generalize to different environments/datasets.

1. Introduction

Depth sensing represents a crucial step for many high-level computer vision applications and stereo is a popular technique to infer dense depth maps from two or more images of the same scene. However, despite a large amount of research in this field, it is still an open problem in particular when facing real applications. This fact clearly stands out dealing with challenging and realistic datasets (Geiger et al., 2013; Menze and Geiger, 2015; Scharstein et al., 2014) on which most stereo algorithms still fail in poorly textured regions, occluded areas and in the presence of other ambiguous elements such as reflective surfaces and so on. Despite Convolutional Neural Networks (CNNs) represent state-of-the-art for disparity estimation, they often require power-hungry GPUs and thus they are not suited for most practical applications, whereas traditional algorithms such as Semi-Global Matching (SGM, by Hirschmuller (2008)) can be effectively implemented on a broad range of devices including low-power systems.

For the reasons outlined, it is also essential to infer the degree of depth uncertainty through *confidence measures*. Such methods have been extensively reviewed and evaluated by Hu and Mordohai (2012) and more recently by Poggi et al. (2017b). Learning-based confidence measures, leveraging random-forests or CNNs, enabled to improve results achieved by *traditional* measures significantly. Compared to CNN methods, approaches based on random forests are potentially faster although most of them less effective. According to Poggi et al. (2017b) learning-based methods processing cues in the disparity domain outperform in most cases approaches based on cost volume analysis. Moreover, such latter input cue is not always available, for instance when dealing with *commercial-off-the-shelf* (COTS) stereo cameras such as RealSense or Zed camera providing as output only a single disparity map. Therefore, confidence measures inferred from features computed in the disparity domain only are highly desirable.

In addition to outlier detection, confidence measures also proved to be an excellent cue to improve the accuracy of traditional, yet very popular in many practical applications, stereo algorithms. Such a strategy is particularly appealing when hardware requirements or training issues preclude the deployment of deep networks for depth estimation.

This paper describes a novel methodology, preliminarily proposed in (Poggi and Mattoccia, 2016b), to infer a confidence measure by feeding a random forest classifier with hand-crafted features extracted, in constant time, in the disparity domain. Figure 1 depicts an overview of our strategy: different features are computed on patches of increasing size to obtain meaningful information fed to a random-forest classifier. Moreover, we also propose an effective strategy to improve the accuracy of SGM by applying a *smarter* scanline aggregation step,

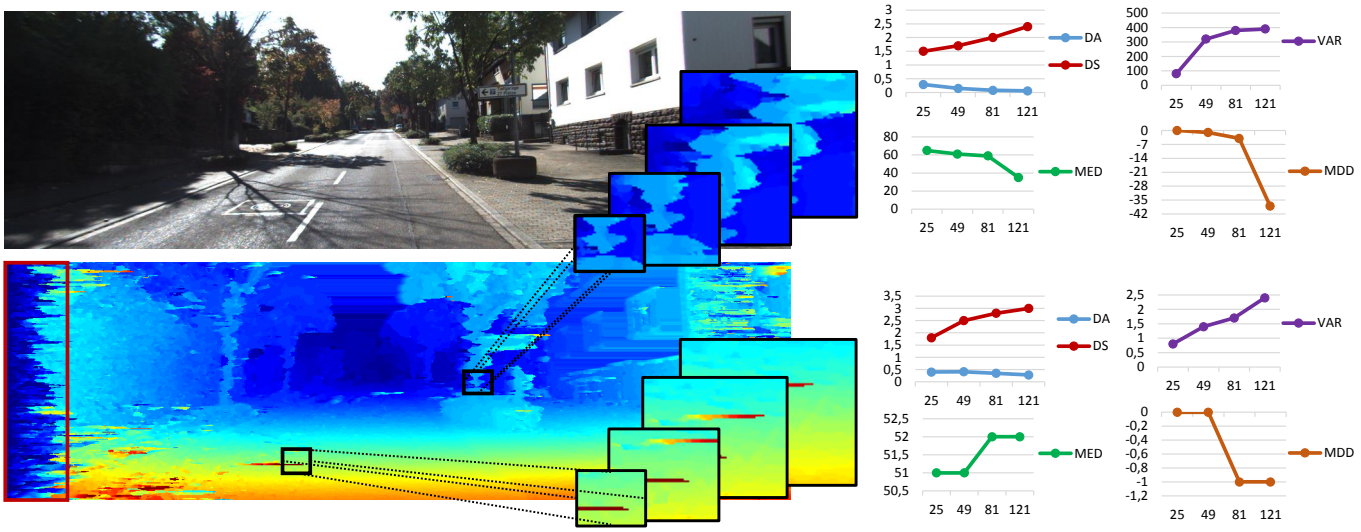

Fig. 1. Overview of the proposed features extraction strategy. Given a disparity map related to the reference image, a confidence measure is learned from features computed in constant time on patches of increasing size.

namely smart-SGM (sSGM). We assess the performance of the proposed confidence estimation methodology on three stereo datasets (KITTI 2012, 2015 and Middlebury v3) and three well-known and popular stereo algorithms (CENSUS, MC-CNN and SGM), as well as a deep stereo network, PSM-Net by Chang and Chen (2018), characterized by a substantially different accuracy. We also profoundly analyze how well the proposed confidence strategy generalizes across different datasets. Purposely, we cross-evaluate confidence measures training on KITTI and testing to Middlebury and vice-versa. Moreover, on the same datasets, we extensively evaluate the performance of the proposed smart aggregation strategy as well as of learning-based variants of SGM, proving the superiority of our proposal.

2. Related Work

In this section, we review the literature concerned with stereo matching, confidence measures and disparity refinement methods since all these fields are related to our work.

In the taxonomy proposed by (Scharstein and Szeliski, 2002), stereo algorithms are 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 and 4) disparity refinement. The SGM algorithm (Hirschmuller, 2008) represents a good trade-off between accuracy and execution time. It independently enforces a *smoothness* constraint on multiple paths employing the Scanline Optimization (SO) algorithm (Scharstein and Szeliski, 2002), summing up each contribution and assigning disparity according to the *Winner-Take-All* (WTA) strategy. Due to its relevance in this field and its favorable computational structure, SGM has been implemented on almost any computing architectures such as GPUs by Zbontar and LeCun (2016), FPGAs by Banz et al. (2010); Gehrig et al. (2009) and other embedded devices whereas mapping end-to-end deep stereo networks would have been hardly feasible in most of the same target devices. Each single SO at the core of SGM is extremely fast but frequently leads to *streaking* artifacts. SGM partially attenuates this issue by summing up independent optimization over multiple paths. Banz et al. (2012) tackled such problem by carefully tuning the smoothing penalties according to the image content, Spangenberg et al. (2013) proposed *weighted*-SGM aimed at adapting the cost of each path according to its fitting with the surface normal while Facciolo et al. (2015) adopted a *more global* strategy.

Although several matching cost functions have been proposed (Hirschmller, 2007), CNN-based methods (Zbontar and LeCun, 2016; Chen et al., 2015; Luo et al., 2016; Shaked and Wolf, 2017; Gidaris and Komodakis, 2017) recently outperformed conventional ones. Further developments in this field lead to *end-to-end* networks able to infer a dense disparity map from a stereo pair without deploying the conventional steps highlighted by Scharstein and Szeliski (2002). In particular, Mayer et al. (2016) proposed DispNet, a fast and accurate architecture achieving quite accurate results at 15+ fps on a GPU. Following this strategy, Kendall et al. (2017) and Pang et al. (2017) proposed 3D convolutions and a multi-staged architecture respectively. Nowadays end-to-end CNNs represents the undisputed state-of-the-art on KITTI (Chang and Chen, 2018; Liang et al., 2018; Tonioni et al., 2019; Guo et al., 2019; Zhang et al., 2019; Poggi et al., 2019), despite their hardware requirements make them unsuited to most practical applications.

Strictly linked to stereo algorithms are confidence measures. *Conventional* approaches have been initially reviewed and evaluated by Hu and Mordohai (2012). Eventually, it has been shown that combining multiple confidence measures and hand-crafted features within random forest frameworks yields significant improvements (Haeusler et al., 2013; Spyropoulos et al., 2014; Park and Yoon, 2015). A significant departure from this strategy was proposed by our previous work (Poggi and Mattoccia, 2016b) computing features in the disparity domain only. Deep-learning played an important role as well enabling to infer

accurate confidence measures by processing patches extracted from the disparity map (Poggi and Mattoccia, 2016c; Seki and Pollefeys, 2016) or global cues extracted from the same domain (Tosi et al., 2018), exploiting local consistency of confidence maps (Poggi and Mattoccia, 2017) and combining multiple confidence measures (Poggi et al., 2017a). Given the fast progress in this field, Poggi et al. (2017b) exhaustively reviewed and evaluated conventional and learning-based confidence estimation strategies, highlighting that methods working in the disparity domain are typically more effective than those relying on features computed from the cost volume.

Confidence measures have been deployed for several purposes: to improve stereo accuracy detecting reliable ground control points (Spyropoulos et al., 2014; Spyropoulos and Mordohai, 2016), to smooth the cost curve (Park and Yoon, 2015), to improve SGM (Poggi and Mattoccia, 2016b; Seki and Pollefeys, 2016) or to fuse multiple depth maps computed by different stereo algorithms (Poggi and Mattoccia, 2016a) or depth sensors (Marin et al., 2016). Moreover, confidence estimation has been used to replace depth labels when adapting deep stereo networks (Tonioni et al., 2017) to unseen environments and for the training of learning-based confidence measures (Tosi et al., 2017).

3. Learning confidence measures in the disparity domain

Combining confidence measures and hand-crafted features with random forest classifiers proved to be very useful for accurate confidence prediction. However, such methods compute cues from the cost volume thus making them not suited for confidence prediction when only a disparity map is given. For instance, such circumstance occurs when COTS devices compute the disparity map, the source code is not provided, the cost volume used to compute the disparity map is no longer available or the disparity map is computed remotely and thus sending the huge cost volume is not feasible. Therefore, our proposal is a significant departure from previous methods (Haeusler et al., 2013; Spyropoulos et al., 2014; Park and Yoon, 2015) being a random forest classifier fed with features uniquely extracted in the reference disparity domain and in constant time. Moreover, it not only outperforms methods based on random forests but it also compares favorably to more complex methods based on CNNs.

3.1. Constant time features inferred from the disparity map

We argue that the local behavior of disparity assignments alone provides powerful enough cues to infer match reliability. For instance, a pixel sharing the same disparity value with a large number of neighboring pixels is more likely to be correct than one sharing it with a few. This cue is particularly useful on smooth or planar surfaces when the change of disparity within nearby pixels is small. Furthermore, many different disparity assignments to nearby pixels may suggest the evidence of a noisy disparity pattern. Therefore, following these observations, for a pixel $p(x, y)$ belonging to the input disparity map $\mathcal{D}$ we encode the local behavior of its neighborhoods into a pool of features computed in constant time at different scales. For simplicity's sake, we omit from now on pixel coordinates (x, y). Thus, given a local window $\mathcal{W}$ centered on p, we define $\mathcal{H}_\mathcal{W}$ as the histogram encoding the distribution of disparity values $d \in [0, d_{max}]$ in $\mathcal{W}$ and we refer to $|\mathcal{W}|$ as its cardinality

$$|\mathcal{W}| = \sum_{d \in [0, d_{max}]} \mathcal{H}(d)_\mathcal{W} \tag{1}$$

Given this notation, we define the following features:

Disparity agreement (DA) encodes the number of neighbors with the same disparity $d(p)$ of the central pixel p:

$$DA(p)_\mathcal{W} = \mathcal{H}(d(p))_\mathcal{W} \tag{2}$$

A larger amount of pixels sharing the same disparity with p encodes a higher likelihood of correctness compared to pixels with smaller support from neighbors.

Disparity scattering (DS) encodes how many different disparity hypotheses appear in the neighborhood of p:

$$DS_p^N = -\log \frac{\sum_{d \in [0, d_{max}]} 1 - \delta_{(\mathcal{H}(d)_\mathcal{W})(0)}}{|\mathcal{W}|} \tag{3}$$

where δ is Kronecker delta function, 1 when no pixel has disparity equal to d (i.e., when $\mathcal{H}(d)_\mathcal{W}$ is 0). According to such definition, a patch of $|\mathcal{W}|$ pixels in complete disagreement with $d(p)$ yields a DS value equal to zero. The lower is the number of different hypotheses within $\mathcal{W}$, the higher is the DS score.

Median disparity (MED) encodes the median of the distribution of disparity hypotheses within the patch $\mathcal{W}$ centered in p:

$$MED(p)_\mathcal{W} = median(\mathcal{H}(d)_\mathcal{W}) \tag{4}$$

Variance of the disparity values (VAR) (Park and Yoon, 2015) encodes the *sparseness* of disparity assignments within $\mathcal{W}$:

$$VAR(p)_\mathcal{W} = \frac{1}{|\mathcal{H}_\mathcal{W}|} \sum_{q \in W} (d(q) - \mu_{|\mathcal{H}_\mathcal{W}|})^2 \tag{5}$$

with

$$\mu_{|\mathcal{H}_\mathcal{W}|} = \frac{1}{|\mathcal{H}_\mathcal{W}|} \sum_{q \in \mathcal{W}} d(q) \tag{6}$$

Disparity deviation from median (MDD) (Spyropoulos et al., 2014) is the negative of the absolute difference between the disparity in p and the median disparity value within patch $\mathcal{W}$:

$$MDD(p) = -\mid d(p) - MED(p)_\mathcal{W} \mid \tag{7}$$

Distance from left border (DLB) (Spyropoulos et al., 2014) assigns lower reliability to pixels closer to the left border where not all the potential candidates are available in the target image. DLB is encoded by the x pixel coordinate, truncated to d_{max}:

$$DLB(p) = \min(x, d_{max}) \tag{8}$$

Uniqueness constraint (UC) (Di Stefano et al., 2004) detects violation of the assumption that each pixel on the target

image can have, at most, one correspondence in the reference image:

$$UC(\mathbf{p}) = \begin{cases} 0, & \text{if } |Q| \neq 0 \\ 1, & \text{otherwise} \end{cases} \quad (9)$$

with Q the set of pixels on reference image having the same destination in the target image.

Other cues computed in the image or disparity domain (Poggi et al., 2017b) such as horizontal gradient magnitude (Haeusler et al., 2013), distance to edges or distance to discontinuities (Spyropoulos et al., 2014) did not yield significant improvements in our experiments, while other traditional measures (Poggi et al., 2017b) are not compliant with our input domain (e.g., LRC requires the target disparity map, usually not available from COTS stereo cameras). By deploying histogram-based optimization (Kass and Solomon, 2010) or box-filtering techniques, all features are computed in constant time regardless of the patch size.

O1 and O2 confidence measures. In this section we outline the feature vectors, fed to a random forest classifier trained in regression mode, to infer two confidence measures referred to as O1, as originally proposed by Poggi and Mattoccia (2016b), and O2 proposed in this paper. Local cues computed at different scales allow to effectively discriminate disparity distributions peculiar of specific image regions such as planar surfaces and discontinuities. Therefore, we include multiple instances of DA, DS, MED, VAR and MDD features computed on patches of increasing size as depicted in Figure 1.

For O1 (Poggi and Mattoccia, 2016b) we define a set of local windows $\mathcal{W}_i$ of size $(3 + 2i) \times (3 + 2i)$ (i.e., for $i \in [1..4]$ we have 5×5, 7×7, 9×9 and 11×11 windows) obtaining a feature vector f_{O1} of 20 elements including DA, DS, MED, VAR and MDD at four scales $f_{O1} = \{DA(p)_{\mathcal{W}_{1..4}}, DS(p)_{\mathcal{W}_{1..4}}, MED(p)_{\mathcal{W}_{1..4}}, VAR(p)_{\mathcal{W}_{1..4}}, MDD(p)_{\mathcal{W}_{1..4}}\}$. We also propose an extended feature vector f_{O2} made of 47 features including additional cues DLB, UC and a larger number of scales with size up to 21×21 (e.g., the set of windows $\mathcal{W}_i$ for $i \in [1..9]$). It allows for a larger receptive field compared to f_{O1}, that proved to be effective for deep learning methods (Poggi and Mattoccia, 2016c) and (Tosi et al., 2018) as well. We refer to this methods as O2 and its feature vector is $f_{O2} = \{DA(p)_{\mathcal{W}_{1..9}}, DS(p)_{\mathcal{W}_{1..9}}, MED(p)_{\mathcal{W}_{1..9}}, VAR(p)_{\mathcal{W}_{1..9}}, MDD(p)_{\mathcal{W}_{1..9}}, DLB, UC\}$. This configuration was chosen after exhaustively experimenting with different scales and their combinations. In particular, we found out that considering neighborhoods larger than 21×21 does not increase accuracy significantly (or may even lead to poor results) while removing some $\mathcal{W}_i$ always yields worse performance compared to f_{O2}. Moreover, including DLB and UC to f_{O2} allows, in some circumstances, to further improve the overall effectiveness. In particular, DLB near the left border while UC near depth discontinuities.

4. Smart Semi Global Matching

SGM (Hirschmuller, 2008) represents an excellent trade-off between accuracy and computational complexity and consequently prevalent in most practical applications. Moreover,

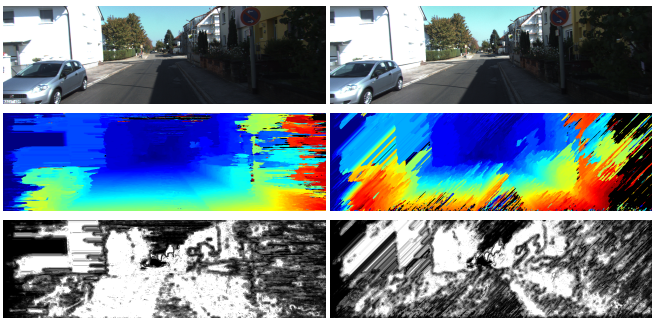

Fig. 2. Examples of streaking artifacts on a stereo pair of the KITTI dataset. Top: stereo pair. Middle: disparity maps computed by SO algorithm along two paths. Bottom: corresponding O2 confidence maps.

many top-performing algorithms still rely on such a method to obtain compelling results on standard evaluation datasets (Geiger et al., 2013; Menze and Geiger, 2015; Scharstein et al., 2014). For each pixel p, SGM combines the outcome of multiple energy minimizations computed by independent SO (Scharstein and Szeliski, 2002) instances on different paths $s \in S$, typically 8 or 16 according to Hirschmuller (2008). 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 $E_s(p, d)$,

$$E_s(p,d) = c(p,d) + \min\Big\{E_s(p',d), E_s(p',d \pm 1) + P1, \min_{i \notin [d-1,d+1]}(E_s(p',i) + P2)\Big\} - \min_{i \in [0,d_{max}]}(E_s(p',i)) \quad (10)$$

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 pixels along epipolar lines. Parameters $P1$ and $P2$ ($P1 < P2$) in (10) enforce *smoothing* by penalizing disparity variations along the path. According to Hirschmller (2007), among the many cost functions proposed for stereo *non-parametric* approaches such as *census* perform very well in challenging environments (Geiger et al., 2013; Menze and Geiger, 2015; Scharstein et al., 2014). Compared to *global* approaches enforcing a smoothness term on a grid (i.e., 2D domain), SO is in most cases much less computationally demanding. However, it is well-known that SO is prone to streaking artifacts along the path direction, as shown in Figure 2. SGM softens this effect by summing up, for each pixel p, the results yielded by multiple SO instances.Finally, SGM infers disparity for pixel p according to WTA.

The proposed sSGM (Poggi and Mattoccia, 2016b) aims at tackling such issues by learning a *smarter* aggregation strategy driven by the analysis of SOs computed along each path. That is, given a pixel p, sSGM 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 a confidence measure. Specifically, for each p, we aggregate the SO costs according to the following weighted sum:

$$E^*(p,d) = \sum_{s \in S} C_s(p) E_s(p,d) \quad (11)$$

As reported in the experimental results, sSGM coupled with O1 or O2 enables to significantly outperforms any other SGM variant known in the literature leveraging on confidence measures.

5. Experimental results

In this section, we report extensive experimental results concerning our proposals. Firstly, we evaluate O1 and O2 confidence measures on KITTI 2012, KITTI 2015 and Middlebury v3 datasets with three popular stereo algorithms – CENSUS, MC-CNN and SGM – and a deep stereo network – PSM-Net – characterized by significantly different accuracy and strategies to tackle the correspondence problem. Their performance is compared to state-of-the-art confidence measures based on random forest, deep learning and two conventional (i.e., not learning-based) strategies PKRN and LRD. Moreover, we also provide a detailed analysis concerning how O1 and O2 behave across datasets depicting very different environments (e.g., KITTIs vs Middlebury datasets). On the same datasets, we also provide exhaustive experimental concerning sSGM and state-of-the-art machine learning variants of SGM.

5.1. Evaluation of confidence measures

A well-established protocol to evaluate the effectiveness of confidence measures consists in Area Under the Curve (AUC) analysis proposed by Hu and Mordohai (2012), computed under the curve plotted by sampling pixels in descending order of confidence. The optimal AUC score is obtained as function of the percentage ε of outliers having a disparity error larger than τ, as $\varepsilon + (1-\varepsilon)\ln(1-\varepsilon)$. A lower AUC corresponds to a better confidence estimation capability. For each dataset and each stereo algorithm, we report average AUC values on the whole dataset to provide a synthetic score over a large set of images as common in this field (Poggi et al., 2017b). Since the ground-truth disparity is required for testing purposes, we always use the training set of each considered dataset. For instance, with KITTI 2015 we mean the KITTI 2015 training dataset. Regarding the stereo algorithms deployed in our evaluation, we consider three popular methods characterized by different degrees of accuracy. For all the algorithms the disparity selection method is WTA. The three algorithms are respectively CENSUS (Zabih and Woodfill, 1994), obtained aggregating on 5×5 support windows the pointwise Hamming distance between 5×5 census transformed images; SGM with eight scanlines and aggregated matching costs computed as described for CENSUS. P1 and P2 are set, respectively, to 0.2 and 0.5 (Poggi et al., 2017b); MC-CNN by Zbontar and LeCun (2016), a Siamese CNN in charge of computing matching costs by processing 9×9 patches. For our experiments, we deployed MC-CNN-*fst* using weights provided by the authors. Compared to MC-CNN-*acrt* this version yields a slightly higher error rate (about 2%), but it is much faster (about 100 times). Concerning PSM-Net, we used the code made available by the authors Chang and Chen (2018) and weights trained on SceneFlow synthetic dataset Mayer et al. (2016) and further fine-tuned on KITTI 2012. This deep network leads to extremely accurate disparity estimation on KITTI 2012 itself, slightly less on KITTI 2015 and quite worse on Middlebury. Moreover, we point out that with PSM-Net, we can only evaluate confidence measures processing the left disparity map and optionally the reference image. This constraint occurs because the cost volume representation processed by the network (and required by other random forest approaches) is quite different from the one of traditional stereo algorithms and the right disparity map is not available at all.

We trained O1 and O2 confidence measures with twenty stereo pairs from the KITTI 2012 dataset, following the protocol outlined by Poggi et al. (2017b), thus providing to the random forest about 2.7 million samples. We trained according to the same protocol confidence measures based on random forest (Haeusler et al., 2013; Spyropoulos et al., 2014; Park and Yoon, 2015) respectively referred to as ENS, GCP an LEV as well as CNN-based approaches (Poggi and Mattoccia, 2016c; Seki and Pollefeys, 2016; Tosi et al., 2018) respectively referred to as CCNN, PBCP and LGC. Moreover, the same measures trained on KITTI 2012 are cross-validated on KITTI 2015 and Middlebury v3 (on this latter, we process quarter resolution images to keep the same disparity range thus allowing a fair comparison). Such evaluation allows perceiving how confidence measures behave on environments different from those learned from training samples, a circumstance often found in practical applications. Purposely, we will also report the gap between this setup and training on the target environment.

Table 1 reports results concerning the AUC evaluation on the three different datasets (excluding frames involved in training for KITTI 2012), multiplied by a factor 10^2 to improve readability. Each sub-table contains experiments using the three stereo algorithms considered. For completeness, we include in our evaluation PKRN and LRD as the baseline for conventional confidence measures. For KITTI 2012 (left), average AUCs have been computed on 174 stereo pairs, the first 20 out of 194 images have been deployed for training (Poggi et al., 2017b), while on KITTI 2015 (center) we average over the entire training set of 200 images. The error threshold is set to $\tau = 3$, compliant with the KITTI on-line evaluation benchmark, to distinguish inliers from outliers. As already highlighted in the literature, O1 yields good overall performance. It always outperforms other methods based on random forest and cost volume analysis ENS, GCP, LEV (this latter method has slightly better AUC in one case, SGM on KITTI 2012) proving the effectiveness of extracting features in the disparity domain. O1 also performs similarly to CNN-based method PBCP, being however outperformed by such method processing the noisy disparity maps provided by CENSUS. On the other hand, O2 always outperforms O1 and PBCP. Moreover, it is substantially equivalent to CCNN when dealing with smooth SGM disparity maps. In this latter case, the larger receptive field of O2 allows for a more effective confidence estimation. Concerning results using PSM-Net, we point out the extremely low optimal AUC. Although effective, all strategies arachieves AUC scores quite far from the optimal (about one order higher on KITTI 2012 and

	KITTI 2012				KITTI 2015				Middlebury v3			
Algorithm	CENSUS	SGM	MC-CNN	PSM-Net	CENSUS	SGM	MC-CNN	PSM-Net	CENSUS	SGM	MC-CNN	PSM-Net
PKRN	22.99 (10)	9.00 (10)	9.85 (10)	-	22.04 (10)	7.98 (10)	9.86 (10)	-	17.54 (9)	10.97 (8)	11.08 (10)	-
LRD	19.46 (9)	8.77 (9)	7.48 (9)	-	18.25 (9)	7.35 (9)	7.12 (9)	-	15.19 (8)	12.18 (10)	9.85 (9)	-
ENS	16.44 (8)	7.63 (8)	4.42 (8)	-	15.10 (8)	7.03 (8)	4.62 (8)	-	19.35 (10)	11.93 (9)	9.66 (8)	-
GCP	15.13 (7)	4.39 (7)	4.15 (7)	-	13.96 (7)	4.04 (7)	4.37 (7)	-	13.57 (7)	10.30 (7)	8.08 (7)	-
LEV	14.27 (6)	3.58 (4)	3.48 (6)	-	13.17 (6)	3.34 (6)	3.64 (6)	-	12.97 (6)	7.37 (5)	7.26 (6)	-
O1	13.09 (5)	3.65 (5)	3.17 (4)	0.43 (4)	11.28 (5)	3.23 (4)	3.24 (4)	1.23 (4)	12.11 (4)	6.09 (3)	6.80 (4)	10.80 (4)
O2	12.93 (3)	3.43 (2)	3.16 (3)	0.39 (3)	11.21 (3)	3.06 (3)	3.16 (3)	1.18 (3)	12.35 (5)	7.84 (6)	6.88 (5)	**10.35 (1)**
PBCP	12.93 (4)	3.68 (6)	3.21 (5)	-	11.27 (4)	3.33 (5)	3.33 (5)	-	11.60 (3)	6.11 (4)	6.52 (3)	-
CCNN	12.23 (2)	3.58 (3)	2.97 (2)	0.31 (2)	10.31 (2)	3.03 (2)	2.97 (2)	1.07 (2)	11.28 (2)	5.95 (2)	6.37 (2)	10.39 (3)
LGC	**11.76 (1)**	**2.78 (1)**	**2.75 (1)**	**0.27 (1)**	**10.04 (1)**	**2.78 (1)**	**1.90 (1)**	**1.04 (1)**	**11.09 (1)**	**6.16 (1)**	**6.09 (1)**	10.36 (2)
Optimal	10.67	2.27	2.31	0.03	8.84	1.84	2.13	0.37	8.99	4.31	4.59	2.28

Table 1. Average AUC on KITTI 2012 (left), KITTI 2015 (center) and Middlebury v3 (right). For each confidence measure each column reports average AUC and rank with CENSUS, SGM, MC-CNN and PSM-Net. Bottom row, optimal AUCs.

3× higher on KITTI 2015). As for traditional algorithms, O1 and O2 are competitive with deep learning solutions, although slightly worse. Finally, the LGC network consistently ranks first for each algorithm and both KITTI datasets thanks to its more complex global reasoning.

On the Middlebury v3 dataset (right), we deal with image content related to indoor environments in order to assess how well a confidence measure performs when dealing with data entirely different from that analyzed during the training phase. In this case, we set $\tau = 1$. Once again we can notice that O1 surmounts any conventional confidence measure as well as any method based on random-forest highlighting once again the effectiveness of extracting features in the disparity domain. Moreover, it also outperforms PBCP with SGM. On the other hand, the table also shows that O2 always has worse performance compared to O1 although O2 is always more effective than ENS, GCP and, excluding SGM, LEV. Concerning PSM-Net, we can notice how its accuracy is lower on Middlebury because of the very different image content compared to KITTI 2012, used for fine-tuning the network. In particular, it also seems much more challenging for the confidence measures to find outliers, achieving AUC scores 5× higher compared to KITTI 2015 than optimal values, while the gap is lower for the experiments with traditional stereo algorithms. It is worth noting that in this experiment, O2 turns out even more accurate than deep learning approaches.

The outcomes of Table 1 indicate that O2 is potentially more effective than O1 but, at the same time, less capable of generalizing to new environments. This aspect will be further discussed in the remainder. Two leading causes can explain the different behaviors of O2 and O1: the more substantial amount of support $\mathcal{W}_i$ and the additional features DLB and UC. Thus, we trained and tested an ablated version of O2 excluding these latter two features. With this setting, O2 achieves average AUC values of 12.99, 3.49, 3.16 and 0.40 respectively, with CENSUS, SGM, MC-CNN and PSM-Net on KITTI 2012 slightly worse than those obtained with the full feature vector f_{O2} reported in Table 1. With the same four methods a similar behaviour is also confirmed on KITTI 2015 – average AUC values of 11.23, 3.08, 3.17 and 1.19 respectively – and Middlebury v3 – average AUC values of 12.37, 7.90, 6.90 and 10.39 respectively – as can be inferred comparing such results with those reported in Table 1. This analysis highlights that DLB and UC positively contribute to the overall O2 performance. However, the main difference between the outcome of O1 and O2 is mostly given by the different amount of support $\mathcal{W}_i$ included in the feature vector f_{O2}.

5.2. Runtime analysis

Concerning the execution time, with a KITTI image at 1241× 376 resolution, using a single core on a standard PC our unoptimized code implemented in C++ and OpenCV takes few milliseconds for PKRN and LRD, about 30 sec for ENS, 5 sec for GCP, 15 sec for LEV and 3.5 sec for O1 and O2. Regarding PBCP, CCNN and LGC, deploying high-end Titan X GPUs the execution times are, respectively, about 0.5, 0.1 and 0.7, rising to about 10 sec on CPU. It is worth to point out that for O1 and O2 the overall computing time is dominated by features computation that can be reduced according to known techniques in the literature, such as histogram optimization (**?**) and box-filtering (Di Stefano et al., 2004), and further accelerated exploiting massive parallelism, provided for instance by GPUs. Moreover, deploying the same strategy, random forest frameworks can achieve significant speedup on GPUs as well as reported in (Grahn et al., 2011).

5.3. Impact of training data and generalization

A relevant aspect concerns how well a confidence measure can generalize across different datasets or, from a different point of view, how it can take advantage from training data similar to that found in the testing environment. The capability to generalize to new environments is a relevant aspect and for this reason we further assess the behavior of O1 and O2 on KITTI 2012 and Middlebury v3 with and without *ad hoc* training on a subset of images similar to the target dataset. That is, for O1 and O2 we report the results achieved with two distinct training configurations: the usual 20 stereo pairs of the KITTI 2012 dataset, as before, and for a fair comparison on an equivalent amount of training samples extracted from Middlebury stereo pairs *Adirondack*, *Vintage*, *Jadeplant*, *Motorcycle*, *Piano*, *Pipes* and *Playroom* not used for the evaluation. With this training configurations, we cross-evaluate O1 and O2 on both Middlebury and KITTI 2012.

Figure 3 reports on top the AUCs on a subset of Middlebury v3 images, from top to bottom with CENSUS, SGM, MC-CNN and PSM-Net. As we can notice, for each stereo algorithm and confidence measure training on Middlebury yields more accurate results compared to those achieved by training

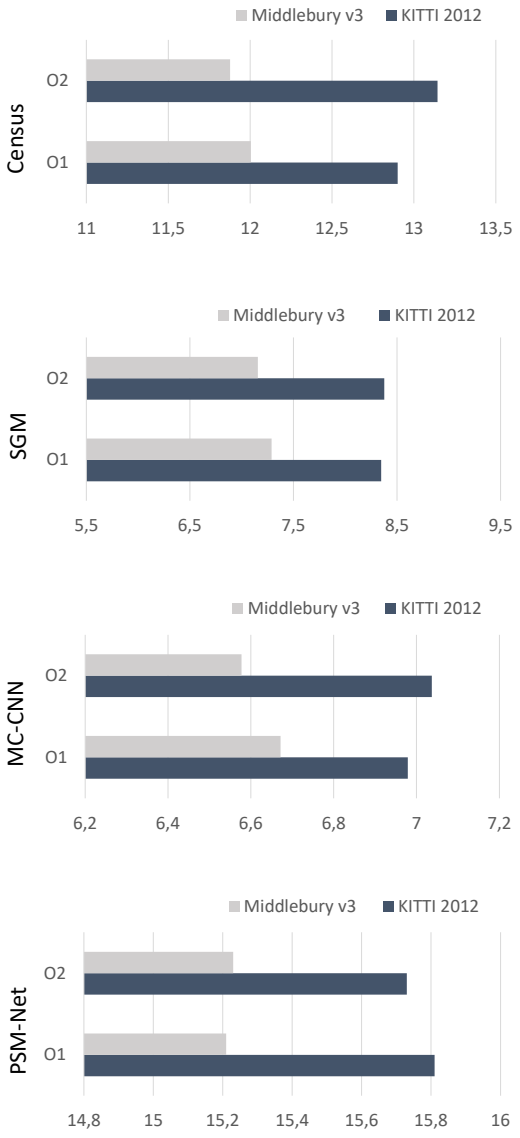

Fig. 3. Generalization capability for O1 and O2. Target dataset is a subset of Middlebury v3. We train on images similar to target domain (bright) or from different datasets (dark).

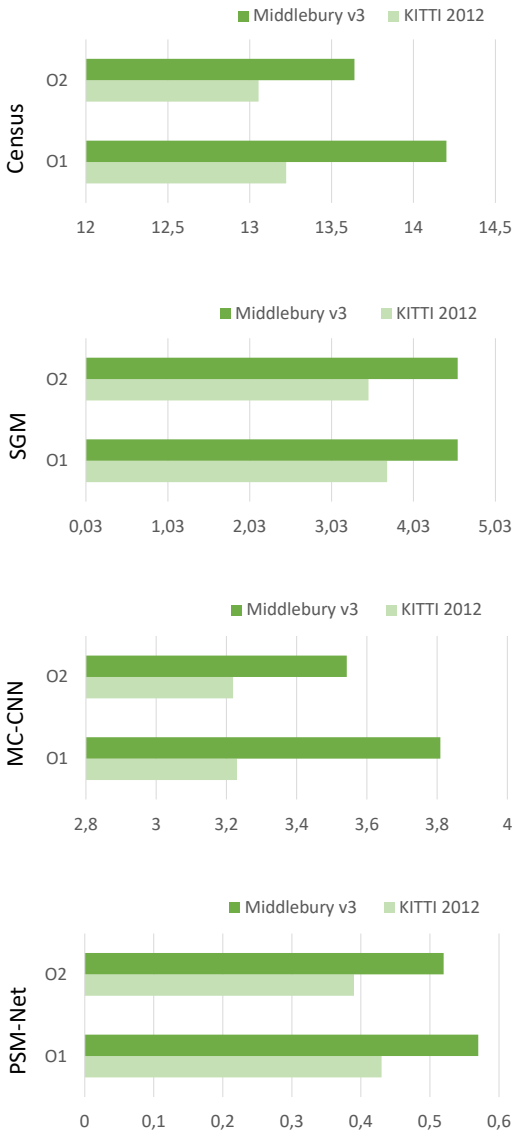

Fig. 4. Generalization capability for O1 and O2. Target dataset is a subset of KITTI 2012. We train on images similar to target domain (bright) or from different datasets (dark).

on KITTI 2012 and this fact is indeed not surprising. However, we can notice how O2 accuracy is notably improved by training on more similar data thus always leading to much better results compared to O1 in the same configuration. This experiment highlights once again the better effectiveness of O2 at the cost, however, of a worse generalization capability across very different scenarios as can be inferred comparing the plots of Figure 3 with the results reported on the right part of Table 1. Nonetheless, we can also notice how, by training on more similar data, the gain in accuracy is quite significant for both confidence measures with all stereo algorithms.

Figure 4 reports the same evaluation by assuming, in this case, KITTI 2012 as target dataset for testing. We compare the AUCs obtained by random forest methods trained on Middlebury and KITTI 2012 datasets. We can see how training on Middlebury allows the O2 measure to achieve better estimation accuracy compared to O1. This outcome can be explained by looking at the nature of the considered datasets: while all samples from KITTI depict similar environments with a low variety of contexts, Middlebury v3 collects many different indoor setups hence providing more heterogeneous training samples as already pointed out by Spyropoulos and Mordohai (2016). Finally, we can notice how the same behaviour is observed, in both cases, when processing disparity map inferred by PSM-Net.

From this analysis we can draw some conclusions: compared to O1, the O2 measure is a more effective confidence estimator but has worse generalization capability across different datasets when *heterogeneous* data are not available for training. Therefore, when the specific environment on which the measure will be deployed is *similar* to the training data, O2 is the best confidence measures based on random forest and it even represents a valid alternative to much more computationally demanding

	KITTI 2012 (bad3)	KITTI 2015 (bad3)	Middlebury v3 (bad1)
SGM (Hirschmuller, 2008)	9.25	8.33	21.92
(Park and Yoon, 2015) + LEV	8.88 (7)	7.92 (7)	21.92 (7)
(Park and Yoon, 2015) + O1	8.87 (6)	7.54 (5)	21.50 (6)
(Park and Yoon, 2015) + O2	8.52 (5)	7.34 (3)	21.27 (5)
(Park and Yoon, 2015) + LGC	8.50 (2)	**7.30 (1)**	21.17 (4)
(Seki and Pollefeys, 2016) + PBCP	9.24 (11)	8.24 (11)	21.90 (8)
(Seki and Pollefeys, 2016) + O1	9.16 (10)	8.25 (10)	21.95 (10)
(Seki and Pollefeys, 2016) + O2	9.05 (9)	8.25 (9)	21.96 (11)
(Seki and Pollefeys, 2016) + LGC	9.04 (8)	8.24 (8)	21.92 (9)
sSGM + O1	8.52 (4)	7.54 (5)	20.78 (3)
sSGM + O2	8.30 (2)	7.37 (4)	20.65 (2)
sSGM + LGC	**8.26 (1)**	7.31 (2)	**20.41 (1)**

Table 2. Comparison of different machine-learning variants of SGM using a variety of measures on KITTI 2012, 2015 and Middlebury v3.

deep learning-based methods PBCP, CCNN and LGC. However, if a more general purpose confidence estimator to deal with scenes quite different from those seen in the training phase is desired, O1 represents a better solution since it is very accurate on average across different datasets and more effective than other learning-based method deploying random forests.

5.4. Evaluation of SGM variants

In this section, we report a comprehensive evaluation of the proposed sSGM algorithm compared to existing machine-learning SGM variants leveraging on confidence measures. In particular, we compare with the cost modulation approach proposed by Park and Yoon (2015) and the dynamic smoothness tuning by Seki and Pollefeys (2016). Table 2 reports the outcome of this evaluation on KITTI 2012, KITTI 2015 and Middlebury v3. We consider the original SGM implementation, the known variants leveraging both the confidence measures they were originally coupled to (i.e., LEV and PBCP) as well as O1 and O2 to be fully comparable with our sSGM. Moreover, we also show how all the variants perform with the LGC measure since it is the best method for outliers detection, as shown in the previous evaluations. For each method we report bad3 for KITTI datasets and bad1 for Middlebury v3, respectively the outliers for $\tau = 3$ and $\tau = 1$. In general, we can notice how using a more effective confidence measure improves the results of each SGM variant, consistently with the outcome of Table 1, with LGC constantly improving the effectiveness of each strategy over the other measures, although marginally, except for Seki and Pollefeys (2016) variant on Middlebury. Moreover, we point out how sSGM outperforms the two competitors most of the times when deploying the same confidence measure, except for KITTI 2015 where Park and Yoon (2015) outperforms it by a negligible margin (0.03%) deploying O2.

6. Conclusions

In this paper, we have proposed a strategy to infer match reliability from features extracted in the disparity domain and in constant time. In contrast to state-of-the-art approaches based on a random forest, our proposal does not require at all the cost volume thus making it deployable with any stereo setup for dense disparity estimation. According to the exhaustive evaluation on standard datasets and algorithms reported, it allows to obtain more accurate confidence estimation compared to methods based on random forests and even compares favorably to much more computationally demanding strategies based on CNNs. We have also introduced a novel strategy to improve stereo accuracy taking advantage of accurate confidence estimators proposed. The proposed sSGM yields overall better accuracy compared to previous variants of SGM in the literature.

Acknowledgements. We gratefully acknowledge the support of NVIDIA Corporation with the donation of a Titan X used for this research.

References

Banz, C., Hesselbarth, S., Flatt, H., Blume, H., Pirsch, P., 2010. Real-time stereo vision system using semi-global matching disparity estimation: Architecture and fpga-implementation., in: ICSAMOS, pp. 93–101.

Banz, C., Pirsch, P., Blume, H., 2012. Evaluation of Penalty Functions for Semi-Global Matching Cost Aggregation. ISPRS - Int. Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences , 1–6.

Chang, J.R., Chen, Y.S., 2018. Pyramid stereo matching network, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Chen, Z., Sun, X., Wang, L., Yu, Y., Huang, C., 2015. A deep visual correspondence embedding model for stereo matching costs, in: CVPR, pp. 972–980.

Di Stefano, L., Marchionni, M., Mattoccia, S., 2004. A fast area-based stereo matching algorithm. Image and Vision Computing 22, 983–1005.

Facciolo, G., de Franchis, C., Meinhardt, E., 2015. Mgm: A significantly more global matching for stereovision, in: BMVC.

Gehrig, S.K., Eberli, F., Meyer, T., 2009. A real-time low-power stereo vision engine using semi-global matching, in: ICVS, pp. 134–143.

Geiger, A., Lenz, P., Stiller, C., Urtasun, R., 2013. Vision meets robotics: The kitti dataset. Int. J. Rob. Res. 32, 1231–1237.

Gidaris, S., Komodakis, N., 2017. Detect, replace, refine: Deep structured prediction for pixel wise labeling, in: CVPR.

Grahn, H., Lavesson, N., Lapajne, M.H., Slat, D., 2011. Cudarf: a cuda-based implementation of random forests, in: 2011 9th IEEE/ACS International Conference on Computer Systems and Applications (AICCSA), IEEE. pp. 95–101.

Guo, X., Yang, K., Yang, W., Wang, X., Li, H., 2019. Group-wise correlation stereo network.

Haeusler, R., Nair, R., Kondermann, D., 2013. Ensemble learning for confidence measures in stereo vision, in: CVPR, pp. 305–312.

Hirschmller, H., 2007. Evaluation of cost functions for stereo matching, in: Computer Vision and Pattern Recognition.

Hirschmuller, H., 2008. Stereo processing by semiglobal matching and mutual information. Trans. on Pattern Analysis and Machine Intelligence 30, 328–341.

Hu, X., Mordohai, P., 2012. A quantitative evaluation of confidence measures for stereo vision. IEEE Trans. on Pattern Analysis and Machine Intelligence , 2121–2133.

Kass, M., Solomon, J., 2010. Smoothed local histogram filters. ACM Trans. Graph. 29, 100:1–100:10.

Kendall, A., Martirosyan, H., Dasgupta, S., Henry, P., Kennedy, R., Bachrach, A., Bry, A., 2017. End-to-end learning of geometry and context for deep stereo regression, in: ICCV.

Liang, Z., Feng, Y., Guo, Y., Liu, H., Chen, W., Qiao, L., Zhou, L., Zhang, J., 2018. Learning for disparity estimation through feature constancy, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Luo, W., Schwing, A.G., Urtasun, R., 2016. Efficient Deep Learning for Stereo Matching, in: CVPR.

Marin, G., Zanuttigh, P., Mattoccia, S., 2016. Reliable fusion of tof and stereo depth driven by confidence measures, in: 14th European Conference on Computer Vision, pp. 386–401.

Mayer, N., Ilg, E., Häusser, P., Fischer, P., Cremers, D., Dosovitskiy, A., Brox, T., 2016. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation, in: CVPR.

Menze, M., Geiger, A., 2015. Object scene flow for autonomous vehicles, in: CVPR.

Pang, J., Sun, W., Ren, J.S., Yang, C., Yan, Q., 2017. Cascade residual learning: A two-stage convolutional neural network for stereo matching, in: ICCV Workshop on Geometry Meets Deep Learning.

Park, M.G., Yoon, K.J., 2015. Leveraging stereo matching with learning-based confidence measures, in: CVPR.

Poggi, M., Mattoccia, S., 2016a. Deep stereo fusion: combining multiple disparity hypotheses with deep-learning, in: 3DV.

Poggi, M., Mattoccia, S., 2016b. Learning a general-purpose confidence measure based on o(1) features and a smarter aggregation strategy for semi global matching, in: 3DV.

Poggi, M., Mattoccia, S., 2016c. Learning from scratch a confidence measure, in: 27th British Conference on Machine Vision.

Poggi, M., Mattoccia, S., 2017. Learning to predict stereo reliability enforcing local consistency of confidence maps, in: CVPR.

Poggi, M., Pallotti, D., Tosi, F., Mattoccia, S., 2019. Guided stereo matching, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

Poggi, M., Tosi, F., Mattoccia, S., 2017a. Even more confident predictions with deep machine-learning, in: 2017 IEEE CVPR Workshops, CVPR Workshops, Honolulu, HI, USA, July 21-26, 2017, pp. 393–401.

Poggi, M., Tosi, F., Mattoccia, S., 2017b. Quantitative evaluation of confidence measures in a machine learning world, in: ICCV.

Scharstein, D., Hirschmller, H., Kitajima, Y., Krathwohl, G., Nesic, N., Wang, X., Westling, P., 2014. High-resolution stereo datasets with subpixel-accurate ground truth., in: GCPR.

Scharstein, D., Szeliski, R., 2002. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. IJCV 47, 7–42.

Seki, A., Pollefeys, M., 2016. Patch based confidence prediction for dense disparity map, in: BMVC.

Shaked, A., Wolf, L., 2017. Improved stereo matching with constant highway networks and reflective confidence learning, in: CVPR.

Spangenberg, R., Langner, T., Rojas, R., 2013. Weighted semi-global matching and center-symmetric census transform for robust driver assistance, in: 15th Computer Analysis of Images and Patterns, pp. 34–41.

Spyropoulos, A., Komodakis, N., Mordohai, P., 2014. Learning to detect ground control points for improving the accuracy of stereo matching., in: Conference on Computer Vision and Pattern Recognition, pp. 1621–1628.

Spyropoulos, A., Mordohai, P., 2016. Correctness prediction, accuracy improvement and generalization of stereo matching using supervised learning. IJCV 118, 300–318.

Tonioni, A., Poggi, M., Mattoccia, S., Di Stefano, L., 2017. Unsupervised adaptation for deep stereo, in: ICCV.

Tonioni, A., Tosi, F., Poggi, M., Mattoccia, S., Di Stefano, L., 2019. Real-time self-adaptive deep stereo.

Tosi, F., Poggi, M., Benincasa, A., Mattoccia, S., 2018. Beyond local reasoning for stereo confidence estimation with deep learning, in: ECCV.

Tosi, F., Poggi, M., Tonioni, A., Di Stefano, L., Mattoccia, S., 2017. Learning confidence measures in the wild, in: BMVC.

Zabih, R., Woodfill, J., 1994. Non-parametric local transforms for computing visual correspondence, in: Third European Conf. on Computer Vision.

Zbontar, J., LeCun, Y., 2016. Stereo matching by training a convolutional neural network to compare image patches. Journal of Machine Learning Research 17, 1–32.

Zhang, F., Prisacariu, V., Yang, R., Torr, P.H., 2019. Ga-net: Guided aggregation net for end-to-end stereo matching, in: CVPR.