Self-adapting confidence estimation for stereo

Self-adapting confidence estimation for stereo

Matteo Poggi Filippo Aleotti Fabio Tosi
Giulio Zaccaroni Stefano Mattoccia

University of Bologna, Viale del Risorgimento 2, Bologna, Italy

Abstract. Estimating the confidence of disparity maps inferred by a stereo algorithm has become a very relevant task in the years, due to the increasing number of applications leveraging such cue. Although self-supervised learning has recently spread across many computer vision tasks, it has been barely considered in the field of confidence estimation. In this paper, we propose a flexible and lightweight solution enabling self-adapting confidence estimation agnostic to the stereo algorithm or network. Our approach relies on the minimum information available in any stereo setup (i.e., the input stereo pair and the output disparity map) to learn an effective confidence measure. This strategy allows us not only a seamless integration with any stereo system, including consumer and industrial devices equipped with undisclosed stereo perception methods, but also, due to its self-adapting capability, for its out-of-the-box deployment in the field. Exhaustive experimental results with different standard datasets support our claims, showing how our solution is the first-ever enabling online learning of accurate confidence estimation for any stereo system and without any requirement for the end-user.

Keywords: stereo matching, confidence, online adaptation

1 Introduction

Stereo is one of the most popular strategies to accurately perceive the 3D structure of the scene through two synchronized cameras and several algorithms, either hand-designed or based on deep neural networks, exist. In many practical applications, alongside with disparity inference, confidence estimation is often performed as well. Purposely, a wide range of methods based either on hand-crafted measures [17] or *learning-based* strategies [39] have been proposed. Recent works [57, 20, 12] showed how state-of-the-art networks processing cues available from any stereo setup, i.e. the input stereo pair and the output disparity map, are substantially equivalent to those processing the entire cost volume [20], further supporting the evidence that the disparity map itself contains sufficient clues to identify outliers as initially proposed in [34, 35]. Such a feature is highly desirable since it potentially paves the way for learning confidence estimation for any stereo camera even without any knowledge about the stereo algorithm/network deployed. This fact is very appealing since it frequently occurs

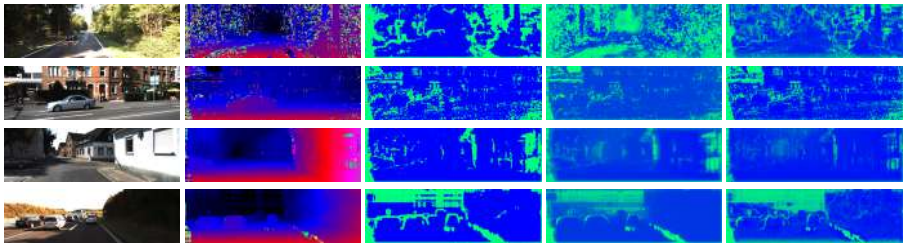

Fig. 1. Self-supervised confidence estimation. From left, reference image, disparity from various algorithms and confidence estimated by self-supervised frameworks [58], [28] and ours. From top to bottom: Census-CBCA, MCCNN-fst-CBCA, Census-SGM and MCCNN-fst-SGM. Color encoding details in the supplementary material.

with most industrial/off-the-shelf (e.g. Stereolabs ZED 2) or consumer devices (e.g. smartphones). Nonetheless, this opportunity was investigated only partially in the literature. Moreover, all these methods are strongly constrained to the need for ground truth depth labels acquired in the target domain. However, since achieving such labels is cumbersome and time-consuming, two self-supervised paradigms have been proposed in the literature [28, 58]. Although these methods proved that confidence estimation could be learned without needing active sensors, they have severe constraints. Individually, [28] requires static stereo sequences while [58] needs access to the *cost volume*, rarely exposed in the case of off-the-shelf stereo sensors or not defined at all in most modern neural networks [26, 22, 56]. As a consequence, both are not thought to handle *adaptation*, required to soften domain-shift issues.

Thus, a solution for out-of-the-box deployment of *self-adaptive* confidence estimation would be highly desirable for many practical applications. A notable example concerns smartphone (e.g. Apple iPhone) nowadays equipped with multiple cameras and undisclosed stereo algorithms/networks deployed for augmented reality or other applications in unpredictable environments.

Therefore in this paper, inspired by recent works performing continuous learning [56, 2] for depth estimation, we propose the first-ever solution for self-adapting a confidence measure unconstrained to the target stereo system. For this purpose, we deploy a novel loss function built upon cues available from the input stereo pair and the output disparity only, needing no additional information to learn/adapt to the sensed environment. Our solution is comparable, and often better, w.r.t known strategies requiring full access to the cost volume [58] or static scenes for training [28], as shown in Fig. 1 on a variety of algorithms.

Extensive experimental results on KITTI, Middlebury 2014, ETH3D and DrivingStereo datasets support the following main claims of our novel confidence estimation paradigm: 1) competitive (often, better) with state-of-the-art when trained in a conventional, offline manner and tested on KITTI; 2) superior generalization capability on other datasets (e.g., Middlebury and ETH3D)

compared to known self-supervised methods; 3) capable of online adaptation, outperforming competitors in unseen environments (e.g., DrivingStereo).

2 Related work

In this section, we review the literature concerning confidence measures and recent trends in stereo matching.

Confidence measures for stereo. Confidence measures have been, at first, reviewed and evaluated in [17] and, more recently, in [39] highlighting that two broad categories exist: *hand-made* and *learned* measures. The former class consists of conventional method computed typically from cost volume analysis such as the ratio between two minima, as in PKR [15], or, as more recently proposed, determining local properties of the disparity map like the number of pixels with the same disparity hypothesis (DA [34]). Concerning learned measures [39], hand-made cues are usually combined and fed as input to a random forest classifier [13, 50, 29, 30, 21, 34, 40] or to a CNN [48, 35, 36, 38, 7, 57, 20, 12] appropriately trained deploying depth labels. Learned methods may require 1) full access to the cost volume to extract hand-made features [13, 50, 29, 30, 21, 38, 36] or process the volume itself [20, 12], 2) disparity maps for both left and right viewpoint [48] or 3) only the input image and its corresponding disparity map [34, 40, 35, 7, 57]. These three requirements translate into harder to softer constraints at deployment, most of them usually not met by off-the-shelf stereo cameras since exposing only the input stereo pair and the output disparity map to the user. Latest works [20, 12] showed that, although a CNN with access to the full cost volume can perform better than networks processing disparity and reference image only, the margin between the two approaches is small and in most cases negligible, at the cost of a much minor versatility of the former.

Applications of confidence measures. In addition to the traditional outliers filtering task, many higher-level applications exploit such cue for different purposes. Again, two main categories exist, acting inside a stereo algorithm or outside it. Belonging to the former, Spyropoulos and Mordohai [50, 52] estimate confidence and detect *ground control points* to improve global optimization. Park and Yoon [29, 30] proposed a confidence-based modulation of the cost volume applied before SGM optimization, Poggi and Mattoccia [34, 40] reduced the streaking effects of the SGM [14] stereo algorithm by using a weighted sum of the scanlines according to a confidence measure. Schonberger et al. [46] act similarly, fusing multiple scanlines of SGM using a random forest classifier. Seki and Pollefeys [48] changed P1 and P2 penalties of SGM dynamically according to the estimated confidence. Methods acting outside the stereo algorithms have been proposed for stereo algorithm fusion [51, 33], sensor fusion [24, 31], and unsupervised adaptation of deep models for stereo matching [53, 54].

Self-supervised confidence estimation. Self-supervised learning has been barely investigated for confidence estimation. Mostegel et al. [28] leverage stereo videos looking at consistencies and contradictions between the different viewpoints of a static scene in order to obtain correct and wrong candidates from

a given stereo algorithm. Tosi et al. [58] instead rely on traditional confidence measures to obtain these two sets according to a consensus among them.

Deep stereo and self-adaptation. At first, CNNs have replaced single steps in the stereo pipeline [44], such as cost computation [63, 4, 23], rapidly converging towards end-to-end solutions estimating dense disparity maps by means of 2D [26, 22, 18, 62, 49, 61, 62] or 3D networks [19, 64, 3, 37, 6]. The latest trend consists of casting disparity estimation as a continuous learning problem, thanks to the self-supervision enabled by image reprojection. First works in this direction are [67, 68], while more recent ones further moved in the direction of real-time continuous adaptation [56, 55] to new environments.

3 Learning a confidence measure out-of-the-box

This work aims at proposing a self-supervised paradigm suited for learning a confidence measure, unconstrained from the specific stereo method deployed and capable of self-adaptation. We first classify stereo systems into different categories according to the data they make available, and then we introduce a novel strategy compatible with all of them.

3.1 Taxonomy of stereo matching systems

In this section, we define three main broad categories of stereo matching solutions, each one characterized by different data made available during deployment. From now on, we will refer to a generic rectified stereo pair as ($\mathcal{I}_L$,$\mathcal{I}_R$), respectively made of left and right images, and to a generic stereo algorithm or deep network as $\mathcal{S}$. In the remainder, to simplify notation, we omit (x, y) coordinates if not strictly necessary.

Black-box models. Given any stereo algorithm processing a stereo pair ($\mathcal{I}_L$,$\mathcal{I}_R$), we define the output disparity map, computed assuming $\mathcal{I}_L$ as the **reference** image, as $\mathcal{D}_L = \mathcal{S}(\mathcal{I}_L, \mathcal{I}_R)$. This image triplet is the minimum amount of data available out of any stereo method, and we define as **black-box** all the systems making available only such cues. Such systems are highly representative of off-the-shelf stereo cameras (e.g., Stereolabs ZED 2) or stereo methods implemented in consumer devices (e.g., Apple iPhones). They, neither allow end-users to access the implementation nor provide explicit ways (APIs) to call for it. For each ($\mathcal{I}_L$,$\mathcal{I}_R$) acquired in the field by the device, they provide the corresponding disparity map typically with undisclosed approaches based either on conventional stereo algorithms or deep networks. Hence, learning confidence measures for these systems is particularly challenging, yet appealing.

Gray-box models. Although black-box systems provide cues available in any stereo system, when explicit calls to the algorithm APIs are exposed, additional cues can be retrieved. Hence, we define a second family of systems for which, although it is given no access to the algorithm implementation or its intermediate data, explicit calls to the method itself are possible (e.g. stereo algorithms provided by pre-compiled libraries). Most deep stereo networks prevent

the deployment of their internal representation since too abstract and substantially unintelligible, e.g. 2D architectures [26, 22, 18, 62, 49, 61]. We define systems belonging to this class as **gray-box**, since multiple calls to $\mathcal{S}$ allow for retrieving additional cues. For instance, it is straightforward to compute the Left to Right Consistency (LRC) of the disparity maps, a popular strategy to obtain a confidence estimator, even if not explicitly provided by $\mathcal{S}$ itself in its original implementation. Given the possibility to call $\mathcal{S}$ two times, consistency checking can be performed analyzing $\mathcal{D}_L$ and a second disparity map, namely $\mathcal{D}_R$ obtained by assuming $\mathcal{I}_R$ as the reference images. Defining $\leftarrow$ the horizontal flipping operator, $\mathcal{D}_R$ is obtained as follows:

$$\mathcal{D}_R = \overleftarrow{\mathcal{S}(\overleftarrow{\mathcal{I}_R}, \overleftarrow{\mathcal{I}_L})} \tag{1}$$

Once obtained $\mathcal{D}_R$, the consistency between the two can be checked as

$$\text{LRC} = |\mathcal{D}_L - \pi(\mathcal{D}_L, \mathcal{D}_R)| < \delta \tag{2}$$

with $\pi(a, b)$ a sampling operator, collecting values at coordinate a from b, and δ a threshold value (usually 1) above which $\mathcal{D}_L$ and $\mathcal{D}_R$ are considered inconsistent. Although less effective than other measures [17], it comes at a lower price.

White-box models. Finally, if the implementation of $\mathcal{S}$ is accessible, additional cues can be sourced by processing intermediate data structures, if meaningful. The preferred one is the cost volume $\mathcal{V}$, containing matching costs $\mathcal{V}(x, y, d)$ for pixels at coordinates (x, y) and any disparity hypothesis $d \in [0, d_{max}]$. This class of systems, referred to as **white-boxes**, enables computation of any confidence measure, either conventional [17] or learning-based [39, 20, 12]. Popular traditional confidence measures obtained from $\mathcal{V}$ are the Peak-Ratio (PKR) and Left-Right Difference (LRD) defined, respectively, as

$$\text{PKR} = \frac{\mathcal{V}(d_{2m})}{\mathcal{V}(d_1)} \quad \text{and} \quad \text{LRD} = \frac{\mathcal{V}(d_2) - \mathcal{V}(d_1)}{\mathcal{V}(d_1) - \min_d \mathcal{V}_R(x - d_1, y, d)} \tag{3}$$

with d_1, d_2 and d_{2m}, respectively, the disparity hypotheses corresponding to the minimum cost, the second minimum and the second local minima [17]. Regarding LRD, given the cost volume $\mathcal{V}_R$ computed assuming $\mathcal{I}_R$ as the reference image, for any pixel (x, y) we sample costs at $(x - d_1, y)$, i.e., from the estimated matching pixel.

Motivations and challenges. Indeed, for the reasons outlined so far, black-box models represent the most challenging, yet general and appealing target when dealing with confidence estimation since their constraints prevent the deployment of most state-of-the-art measures [20, 12], as well as self-supervised strategy existing in the literature [28, 58]. Hence, first and foremost, we aim at devising a general-purpose strategy enabling self-supervised confidence estimation in such constrained settings. As a notable consequence, this fact paves the way to tackle the same task even for state-of-the-art CNNs. Finally, having achieved this goal, out-of-the-box learning of confidence estimation with any stereo setup and self-adaptation in any environment is at hand.

3.2 Self-supervision cues for black-box models

In order to develop a self-supervised strategy suited for any stereo system, it is crucial to identify cues that are effective to source a robust supervision signal. According to the previous discussion, in the case of black-box models, we can rely on $(\mathcal{I}_L, \mathcal{I}_R)$ and $\mathcal{D}_L$ only. In this circumstance, although relevant information is not available compared to other models, we introduce three terms to obtain the desired self-supervised signal from the meagre cues available.

Image reprojection error. In recent literature, several works proved how the reprojection across the two viewpoints available in a rectified stereo pair could be a powerful source of supervision, either for monocular [10, 32, 11] or stereo [66, 56] depth estimation. Specifically, we can reproject $\mathcal{I}_R$ on the reference image coordinates as $\tilde{\mathcal{I}_R} = \pi(\mathcal{D}_L, \mathcal{I}_R)$ Then, the difference between $\mathcal{I}_L$ and warped right view $\tilde{\mathcal{I}_R}$ appearance encodes how correct the reprojection is. To this aim, the most popular choice is a weighted sum between two terms, respectively SSIM [59] and absolute difference.

$$\Delta_{(\mathcal{I}_L, \tilde{\mathcal{I}_R})} = \alpha \cdot (1 - \text{SSIM}(\mathcal{I}_L, \tilde{\mathcal{I}_R})) + (1 - \alpha)|\mathcal{I}_L - \tilde{\mathcal{I}_R}| \tag{4}$$

with α usually tuned to 0.85. The higher it is, the more likely $\mathcal{D}_L$ is wrong. By definition, matching pixels is particularly challenging in ambiguous regions, such as textureless portions of the image. To this aim, we first aim at detecting regions with rich texture, being more likely to be correctly estimated by $\mathcal{S}$, by comparing Δ computed between $(\mathcal{I}_L, \mathcal{I}_R)$ with the one after reprojection as $\mathcal{T} = \Delta_{(\mathcal{I}_L, \mathcal{I}_R)} > \Delta_{(\mathcal{I}_L, \tilde{\mathcal{I}_R})}$. In large ambiguous regions, $\Delta_{(\mathcal{I}_L, \mathcal{I}_R)}$ will result equal (or even minor) than the reprojection error [11], thus identifying pixels on which stereo is prone to errors.

Agreement among neighboring matches. Since most regions of a disparity map should be smooth, variations in nearby pixels should be small except at depth boundaries. As highlighted in [34, 40], $\mathcal{D}_L$ itself allows for the extraction of meaningful cues to assess the quality of disparity assignments. Purposely, we rely on the **disparity agreement** between neighbouring pixels, defined as

$$\text{DA} = \frac{\mathcal{H}_{N \times N}(d_1)}{N \times N} \tag{5}$$

$\mathcal{H}_{N \times N}$ is an histogram encoding, for each pixel (x, y), the number of neighbours in a $N \times N$ window having the same disparity d (in case of subpixel precision, within 1 pixel). In the absence of depth discontinuities, the majority of pixels in the neighbourhood should share the same, or very similar, disparity hypothesis. Hence, we define a second criterion to identify reliable stereo correspondences as $\mathcal{A} = \text{DA} > 0.5$, assuming that more than half of the pixels in the neighbourhood share the same disparity. It is worth noting that this criterion is often not met in the presence of depth boundaries, even in case of correct disparities.

Uniqueness constraint. In an ideal frontal-parallel scene observed by a stereo camera in standard form, for each pixel in $\mathcal{I}_L$ exists at most one match in $\mathcal{I}_R$ and vice-versa. Leveraging this property, known as uniqueness, is particularly useful [5] to detect outliers in occluded regions and represents a reliable

Fig. 2. Effects of different criteria. Given the highlighted region, we show inliers (green) and outliers (red) guesses by using the following cues in multi-modal binary cross-entropy: a) $\mathcal{T}^p, \mathcal{T}^q$ b) $\mathcal{A}^p, \mathcal{A}^q$ c) $\mathcal{U}^p, \mathcal{U}^q$ d) $\mathcal{T}^p, \mathcal{A}^p, \mathcal{U}^p, \mathcal{T}^q$ e) $\mathcal{T}^p, \mathcal{A}^p, \mathcal{U}^p, \mathcal{T}^q, \mathcal{A}^q, \mathcal{U}^q$. For black pixels, the considered configuration gives no guesses.

alternative to LRC and LRD measures, not usable when dealing with black-box models. Uniqueness Constraint (UC) is encoded as

$$\text{UC} = [x - \mathcal{D}_L(x, y)] \notin \bigcup_k [(x + k) - \mathcal{D}_L(x + k, y)] \tag{6}$$

with $k \in [-d^*_{max}, -1] \cup [1, d^*_{max}]$ and $d^*_{max} = d_{max} - \mathcal{D}_L(x, y)$. In other words, the uniqueness for any pixel in $\mathcal{I}_L$ holds if it does not collide in the target image with any other pixel, i.e., not matching the same pixel in $\mathcal{I}_R$ matched by any other. We exploit this property to define our third criterion as $\mathcal{U}$ =UC. We conclude observing that, although effective at detecting mostly occlusions, the uniqueness constraint is often violated in the presence of slanted surfaces.

3.3 Multi-modal Binary Cross Entropy

Given the three criteria outlined above, we revise the traditional binary cross entropy loss to take into account multiple label hypotheses. We refer to this variant as **Multi-modal Binary Cross Entropy** (MBCE), defined as

$$\mathcal{L}_{\text{MBCE}} = -\left[\left(\prod_{p \in \mathcal{P}} p\right) \cdot log(o) + \left(\prod_{q \in \mathcal{Q}} q\right) \cdot log(1 - o)\right] \tag{7}$$

with o the output of the neural network $\in [0, 1]$, i.e. passed through a sigmoid activation, $\mathcal{P}$ and $\mathcal{Q}$ two sets of **proxy labels** derived respectively by a criterion being met or not. For instance, pixels satisfying the first criterion on image reprojection will have labels $\mathcal{T}^p = 1$, $\mathcal{T}^q = 0$ and vice versa when they do not. Unlike traditional binary cross entropy, where a single label y and its counterpart $(1 - y)$ are used, we define disjoint sets of proxies allowing for a flexible configuration of the loss function according to the three criteria described so far. For instance, by setting $\mathcal{P} = [\mathcal{T}^p, \mathcal{A}^p]$ and $\mathcal{Q} = [\mathcal{T}^q]$ we will train the network to detect good matches using image reprojection plus agreement and outliers using the former only. Adding elements to the sets $\mathcal{P}$ and $\mathcal{Q}$ reduces progressively the number of pixels considered correct or wrong, respectively. Fig. 2 shows this, highlighting how combining multiple guesses as in d) and e) for some pixels no

supervision is given when criteria do not match. We will report the impact of this and the different configurations in a thorough ablation study.

4 Experimental results

In this section, we report an exhaustive evaluation to assess the effectiveness of our strategy, referred to as *Out-of-The-Box* (OTB), by conducting three main experiments, respectively: 1) ablation study on the MBCE loss, 2) comparison with self-supervised approaches [28, 58] in a conventional offline training and 3) an evaluation concerning online adaptation of OTB.

4.1 Implementation details

We now report all the details to understand and reproduce our experiments fully. The source code will be made publicly available at the end of the review process.

Evaluation Protocol. To measure the effectiveness of the learned confidence measures, we compute the Area Under Curve (AUC) of the sparsification plots [17, 39, 57, 20]. Given a disparity map, pixels are sorted in increasing order of confidence and gradually removed (e.g., 5% each time) from the disparity map. At each iteration, the error rate is computed over the sparse disparity map as the percentage of pixels having absolute error larger than τ. Plotting the error rate results in a sparsification curve, whose AUC quantitatively assesses the confidence effectiveness (the lower, the better). Optimal AUC is obtained by sampling the pixels in decreasing order of absolute error.

Confidence networks. Since the goal of this work is to define an effective self-supervised strategy suited for online learning rather than proposing a novel architecture, in our experiments, we test our proposal to train existing networks. Purposely, we consider three architectures: CCNN [35], ConfNet and LGC [57] to carry out our experiments because 1) since they process only disparity map and reference image are suited to all methods, from white-box to black-box, 2) according to recent works [20, 12], the most accurate one (LGC) is on par with state-of-the-art networks processing the cost volume and 3) the source code is fully available, conversely to [21, 12]. Moreover, in ConfNet we replaced deconvolutions with bilinear upsampling followed by 3×3 convolutions and processing $\mathcal{D}_L$ only, significantly improving its performance and thus filling most of the gap with CCNN and LGC. We defined a training schedule for each network, kept constant in all experiments. For CCNN, we use batches of 128 patches for 1M iterations, for ConfNet batches of single, full-resolution images for 25K iterations, finally for LGC batches of 128 patches for 300K iterations, starting from pre-trained CCNN and ConfNet models. We trained all networks with SGD optimizer and a constant learning rate of 0.001.

Datasets. We consider five standard datasets: KITTI 2012 [9], KITTI 2015 [27], Middlebury 2014[1] [43], ETH3D [47] and DrivingStereo [60], setting τ respectively to 3, 3, 1, 1 and 3. Being ground truth required to assess performance,

[1] We use the quarter resolution split as in previous works [39, 57, 20]

we refer to the training set of such datasets. To train confidence estimation networks, we select the first 20 images from KITTI 2012 as in [39, 57] for supervised training and the 400 images from the first 20 sequences of the KITTI 2012 multiview extension used in [28, 58] for self-supervised ones. To evaluate the trained confidence networks, we use the remaining 174 images from KITTI 2012 as the validation set and the totality of images available from KITTI 2015 for experiments on environments similar to the training set. Moreover, we also assess their generalization performance on the whole Middlebury 2014 and ETH3D datasets. In these experiments, only the KITTI 2012 images listed above are used for training. Thus, in the evaluation, the networks are transferred without any fine-tuning or adaptation. Finally, to test self-adaptation peculiar of OTB we use a sequence from the DrivingStereo dataset, namely 2018-10-25-07-37, made of about 7K frames.

Stereo algorithms. Following the recent literature [39, 57, 20], we evaluate the effectiveness of our strategy on a variety of stereo algorithms with different degrees of accuracy, in order to highlight how strong is our self-supervised paradigm in the presence of heterogenous disparity maps. We consider four main stereo algorithms deploying the code provided Zbontar and LeCun [63] under different settings. Specifically: Census-CBCA, Census-SGM, MCCNN-fst-CBCA and MCCNN-fst-SGM. The first two rely on a census-based matching cost computation, respectively, optimized by a Cross Based Cost Aggregation (CBCA) strategy [65] and SGM [14]. The latter two replace the census-based matching costs with predictions obtained by MCCNN-fst, for which we use pre-trained weights on KITTI 2012, 2015 and Middlebury provided by the authors and tested on the same datasets. For ETH3D, Middlebury weights have been used. Furthermore, to evaluate the impact of self-adaptation made possible by OTB with a real black-box method, we also consider two recent deep stereo network. We choose MADNet [56] and GANet [64], both trained on synthetic images [26] and then fine-tuned with ground truth on KITTI 2015, because of the availability of trained model and its accuracy-speed trade-off. Since fine-tuned on KITTI, we conduct experiments with MADNet and GANet on DrivingStereo only.

Competitors. We compare the proposed OTB strategy with existing methods proposed by Mostegel et al. [28] (named SELF) and by Tosi et al. [58] (named WILD). The former reasons about contradictions on observations from multiple viewpoints: given a stereo sequence framing a static scene with a moving camera, $\mathcal{D}_L$ and $\mathcal{D}_R$ are computed for each pair, registered and checked for inconsistencies. Since it requires both $\mathcal{D}_L$ and $\mathcal{D}_R$ disparity maps, SELF is suited only for systems belonging to gray-box and white-box categories. Concerning WILD, it requires a pool of six confidence measures extracted from the cost volume to identify inliers and outliers according to heuristic thresholding on the measures. Since it requires access to the cost volume, WILD is suited for white-box algorithms only. In contrast, among other advantages discussed next, it worth stressing that our OTB approach is suited for black-box systems and agnostic to the scene content, in contrast to SELF that requires static scenes.

KITTI 2012

Match cost						Census		MCCNN-fst		Census		MCCNN-fst		Census		MCCNN-fst	
Aggregation						CBCA	SGM	CBCA	SGM	CBCA	SGM	CBCA	SGM	CBCA	SGM	CBCA	SGM
$\Delta_{\mathcal{I}}(\mathcal{I}_L, \tilde{\mathcal{I}_R})$						0.210	0.086	0.165	0.044	0.210	0.086	0.165	0.044	0.210	0.086	0.165	0.044
DA						0.112	0.047	0.063	0.023	0.112	0.047	0.063	0.023	0.112	0.047	0.063	0.023
UC						0.165	0.063	0.123	0.034	0.165	0.063	0.123	0.034	0.165	0.063	0.123	0.034
$\mathcal{T}^p$	$\mathcal{A}^p$	$\mathcal{U}^p$	$\mathcal{T}^q$	$\mathcal{A}^q$	$\mathcal{U}^q$	CCNN				ConfNet				LGC			
✓			✓			0.080	0.045	0.047	0.018	0.077	0.033	0.045	0.014	0.082	0.058	0.046	0.026
	✓			✓		0.105	0.045	0.073	0.023	0.087	0.035	0.049	0.017	0.110	0.040	0.074	0.022
		✓			✓	0.111	0.035	0.087	0.022	0.101	0.038	0.065	0.020	0.114	0.035	0.077	0.020
✓	✓		✓			0.078	0.033	0.050	0.019	0.072	0.030	0.038	0.014	0.075	0.034	0.049	0.023
✓	✓		✓	✓		0.089	0.035	0.059	0.023	0.071	0.027	0.038	0.014	0.082	0.031	0.066	0.020
✓		✓	✓			0.072	0.038	0.053	0.019	0.074	0.028	0.040	0.013	0.070	0.036	0.042	0.016
✓		✓	✓		✓	0.088	0.032	0.075	0.020	0.076	0.030	0.041	0.013	0.084	0.031	0.071	0.017
✓	✓	✓	✓			**0.068**	0.034	**0.046**	0.018	**0.070**	0.028	**0.037**	0.013	**0.068**	0.032	**0.041**	0.016
✓	✓	✓	✓	✓	✓	0.085	**0.029**	0.057	**0.017**	0.071	**0.026**	0.038	**0.012**	0.081	**0.028**	0.050	**0.015**

Middlebury

Match cost						Census		MCCNN-fst		Census		MCCNN-fst		Census		MCCNN-fst	
Aggregation						CBCA	SGM	CBCA	SGM	CBCA	SGM	CBCA	SGM	CBCA	SGM	CBCA	SGM
$\Delta_I(\mathcal{I}_L, \mathcal{I}_R)$						0.190	0.180	0.179	0.134	0.190	0.180	0.179	0.134	0.190	0.180	0.179	0.134
DA						0.161	0.168	0.099	0.087	0.161	0.168	0.099	0.087	0.161	0.168	0.099	0.087
$\mathcal{U}$						0.193	0.188	0.192	0.145	0.193	0.188	0.192	0.145	0.193	0.188	0.192	0.145
$\mathcal{T}^p$	$\mathcal{A}^p$	$\mathcal{U}^p$	$\mathcal{T}^q$	$\mathcal{A}^q$	$\mathcal{U}^q$	CCNN				ConfNet				LGC			
✓	✓	✓	✓			**0.116**	**0.123**	**0.087**	**0.077**	**0.133**	**0.112**	**0.087**	**0.067**	**0.127**	**0.111**	**0.090**	**0.064**
✓	✓	✓	✓	✓	✓	0.153	0.146	0.095	0.081	0.134	0.122	0.095	0.069	0.138	0.142	0.099	0.080

Table 1. Ablation study on the proposed multi-modal binary cross entropy. We report AUC scores for networks trained on KITTI 2012 (20 or 400 images) and tested on KITTI 2012 (174 images, top) and Middlebury (15 images, bottom).

4.2 Ablation study

At first, we study the impact of the different terms in the proposed self-supervised loss function. To this aim, on KITTI 2012 and as for other experiments, we train 9 variants of each network for each of the four stereo algorithms. Then, we evaluate confidences on the KITTI 2012 dataset and, without retraining, on Middlebury 2014. Table 1 collects the outcome of this evaluation, reporting on top results on KITTI 2012 and, at the bottom, on Middlebury. We report as baselines the performance of $\Delta_{(\mathcal{I}_L, \tilde{\mathcal{I}_R})}$, DA and UC. DA is computed on 5×5 windows.

On KITTI (top of the table), we first report the results achieved by training the three networks selecting only one of the three cues used to distinguish between correct and wrong matches, i.e. $[\mathcal{T}^p, \mathcal{T}^q]$, $[\mathcal{A}^p, \mathcal{A}^q]$ and $[\mathcal{U}^p, \mathcal{U}^q]$ configurations. We can notice that each of them outperforms the performance of the corresponding baseline used for supervision. This trend occurs on all the algorithms and for each network, showing the surprisingly robust capacity of the networks to learn how to estimate confidence better than a noisy supervision signal used for training. In general, the models trained on $[\mathcal{T}^p, \mathcal{T}^q]$ outperforms the others, except rare cases (i.e. CCNN and LGC on Census-SGM, outperformed by $[\mathcal{U}^p, \mathcal{U}^q]$ setting). Although effective at detecting textureless and ambiguous regions, the reprojection fails at filtering outliers due to slanted surfaces and occlusions. Thus, we incrementally add a single criterion, i.e. $\mathcal{A}^p$ or $\mathcal{U}^p$ to filter out false positives obtained by $[\mathcal{T}^p, \mathcal{T}^q]$ configuration. We incrementally add, on another configuration, the corresponding negative criterion to remove pixels

	Match cost	Census				MCCNN-fst			
	Aggregation	CBCA		SGM		CBCA		SGM	
	KITTI split	2012	2015	2012	2015	2012	2015	2012	2015
	Badτ %	27.193	22.281	10.330	8.998	18.875	16.926	6.084	6.028
Traditional	LRD	0.096	0.080	0.033	0.032	0.080	0.077	0.017	0.023
	PKR	0.106	0.089	0.028	0.029	0.065	0.062	0.010	0.017
	LRC	0.142	0.113	0.062	0.056	0.103	0.092	0.036	0.041
	$\Delta_{(\mathcal{I}_L,\mathcal{I}_R)}$	0.210	0.175	0.086	0.079	0.165	0.150	0.044	0.041
	DA	0.112	0.090	0.047	0.046	0.063	0.059	0.023	0.028
	UC	0.165	0.131	0.063	0.058	0.123	0.111	0.034	0.037
CCNN	Supervised	0.059	0.046	0.018	0.017	0.031	0.032	0.009	0.012
	WILD [58]	0.076	0.065	**0.026**	**0.026**	0.052	0.047	**0.012**	**0.017**
	SELF [28]	0.076	0.065	0.047	0.046	**0.038**	**0.041**	0.012	0.018
	OTB (Ours)	**0.068**	**0.055**	0.029	0.031	0.046	0.048	0.017	0.022
ConfNet	Supervised	0.061	0.049	0.017	0.016	0.033	0.034	0.006	0.010
	WILD [58]	0.089	0.067	**0.024**	**0.020**	0.054	0.050	**0.010**	**0.016**
	SELF [28]	0.075	0.066	0.024	0.024	0.041	0.044	0.014	0.016
	OTB (Ours)	**0.070**	**0.058**	0.026	0.028	**0.037**	**0.040**	0.012	0.017
LGC	Supervised	0.056	0.044	0.016	0.016	0.029	0.030	0.007	0.010
	WILD [58]	0.089	0.065	**0.026**	**0.025**	0.049	0.045	**0.011**	**0.017**
	SELF [28]	0.089	0.081	0.026	0.026	0.056	0.057	0.020	0.021
	OTB (Ours)	**0.068**	**0.055**	0.028	0.032	**0.041**	**0.044**	0.015	0.019
	Optimal	0.047	0.034	0.008	0.008	0.024	0.022	0.003	0.005

Table 2. Evaluation on KITTI. We report AUC scores for networks trained on KITTI 2012 (20 or 400 images) and tested on 2012 (174 images) and 2015 (200 images).

wrongly categorized as outliers by $\mathcal{T}^q$. In most cases, adding a single criterion to $\mathcal{P}$ is beneficial, while we can notice how introducing negative criteria degrades the performance on CBCA algorithms. This occurs because adding $\mathcal{A}^q$ or $\mathcal{U}^q$ makes textureless regions no longer labelled as outliers, as shown in Fig. 2 left comparing patches d) and e). Finally, adding both $\mathcal{A}^p$ and $\mathcal{U}^p$ produces the best overall results for CBCA methods. By introducing $\mathcal{A}^q$ and $\mathcal{U}^q$ too we obtain better results only on SGM methods, since much more accurate than CBCA ones and thus more false outliers are introduced if $\mathcal{A}^q$ and $\mathcal{U}^q$ are not used, as shown in Fig. 2 right, comparing d) and e).

On the other hand, by testing the best configurations on Middlebury 2014, enabling all the positive criteria and only $\mathcal{T}^q$ for negative allows for better generalization to unseen environments.

4.3 Comparison with offline methods

Having found the best configuration for the $\mathcal{L}_{\text{MBCE}}$ loss, we compare our supervision paradigm with known self-supervised approaches [28, 58]. In our experiments, we obtain proxy labels for SELF and WILD using the code provided by the respective authors. We collect the outcome of these experiments in Tables 2 and 3. We label with different colors methods ranging from **stronger** constraints (need for ground truth) to weaker (ours). For each architecture, stereo algorithm and evaluation set triplet we label in **bold** the best self-supervision approach, while in **red** the couple architecture/self-supervision on an entire evaluation set.

KITTI datasets. Table 2 collects evaluations on the KITTI 2012 and 2015 datasets, respectively, using the 174 validation set from 2012 and the full 2015 set. We point out that all self-supervised strategies outperform traditional measures, reported on top as baselines, such as LRD, PKR, LRC and the cues used in our

	Match cost	Census				MCCNN-fst			
	Aggregation	CBCA		SGM		CBCA		SGM	
	Dataset	Midd	ETH	Midd	ETH	Midd	ETH	Midd	ETH
	Bad1 %	28.701	21.270	26.682	15.471	29.7999	34.279	21.799	12.594
Traditional	LRD	0.117	0.082	0.113	0.059	0.107	0.185	0.075	0.051
	PKR	0.124	0.086	0.112	0.056	0.095	0.181	0.059	0.042
	LRC	0.189	0.135	0.197	0.114	0.188	0.239	0.149	0.091
	$\Delta_{(\mathcal{I}_L,\tilde{\mathcal{I}}_R)}$	0.190	0.162	0.180	0.119	0.179	0.257	0.134	0.097
	DA	0.161	0.119	0.168	0.093	0.099	0.159	0.087	0.047
	UC	0.193	0.148	0.188	0.114	0.192	0.264	0.145	0.096
CCNN	Supervised	0.110	0.096	0.118	0.076	0.079	0.138	0.068	0.046
	WILD [58]	0.136	0.114	0.140	0.086	0.095	0.154	0.081	0.046
	SELF [28]	0.163	0.174	0.217	0.174	0.090	0.147	0.081	0.076
	OTB (Ours)	**0.116**	**0.084**	**0.123**	**0.070**	**0.087**	**0.137**	**0.077**	**0.042**
ConfNet	Supervised	0.121	0.086	0.104	0.063	0.086	0.138	0.062	0.036
	WILD [58]	0.122	0.101	0.117	**0.063**	0.091	0.160	0.073	0.037
	SELF [28]	0.154	0.120	0.121	0.067	0.096	0.172	0.084	0.048
	OTB (Ours)	**0.133**	**0.093**	**0.112**	0.067	**0.087**	**0.138**	**0.067**	**0.035**
LGC	Supervised	0.111	0.080	0.111	0.061	0.083	0.136	0.065	0.040
	WILD [58]	0.136	0.104	0.133	0.082	0.098	0.156	0.084	0.050
	SELF [28]	0.128	0.105	0.117	0.066	0.091	0.154	0.086	0.060
	OTB (Ours)	**0.127**	**0.084**	**0.111**	**0.056**	**0.090**	**0.139**	**0.064**	**0.035**
	Optimal	0.053	0.041	0.046	0.022	0.057	0.103	0.030	0.014

Table 3. Generalization on Middlebury and ETH3D. We report AUC scores for networks trained on KITTI 2012 (20 or 400 images) and tested on Middlebury (15 images) and ETH3D (27 images) without retraining or adaptation.

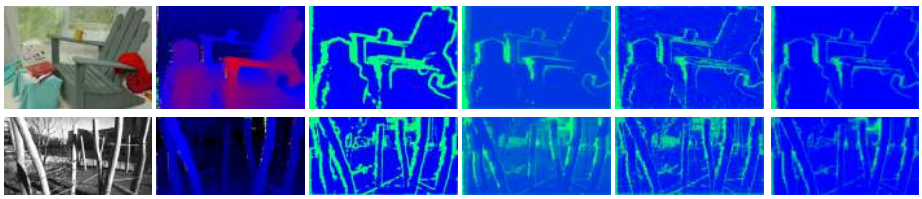

Fig. 3. Qualitative results for generalization. From left: reference image, disparity by MCCNN-fst-SGM, ConfNet trained with [58], [28], our method and ground-truth. On top, Adirondack (Middlebury), at the bottom, Playground_3l (ETH3D).

$\mathcal{L}_{\text{MBCE}}$ loss, struggling only when dealing with the very accurate MCCNN-fst-SGM algorithm. Comparing the different architectures, we can notice how the self-supervised paradigms break the hierarchy (i.e., self-supervised LGC is often outperformed by ConfNet). On Census-CBCA, our strategy always outperforms SELF and WILD when used to train any architecture. The same behaviour is confirmed on MCCNN-fst-CBCA, except for CCNN resulting better with SELF but with the best performance achieved by ConfNet trained with OTB. This outcome highlights the outstanding performance of OTB with noisy algorithms (about 27 and 19% error rates on the validation set), close to **full supervision**. On SGM algorithms, OTB results comparable with SELF and WILD, although sourcing supervision only from images and $\mathcal{D}_L$, thus in a much weaker form compared to the competitors. On three out of four algorithms, ConfNet results to be the most effective architecture when trained in a self-supervised manner.

Generalization on Middlebury and ETH3D. Table 3 reports results on the Middlebury 2014 and ETH3D datasets. We point out that the same networks evaluated so far (trained on KITTI 2012 images) are transferred here

	Traditional			Supervision						
Algorithm	$\Delta_{(\mathcal{I}_L,\tilde{\mathcal{I}}_R)}$	DA	UC	Supervised	WILD [58]	SELF [28]	OTB	**OTB (online)**	Opt.	Badτ %
Census-SGM	0.179	0.106	0.161	0.061	0.067	0.074	0.072	**0.061**	0.029	21.007
MADNet [56]	0.134	0.147	0.152	0.116	-	0.135	0.146	**0.125**	0.021	16.226
GANet [64]	0.046	0.071	0.061	0.044	-	0.050	0.050	**0.046**	0.007	7.247

Table 4. Self-adaptation. We report AUC scores for networks trained on KITTI 2012 (20 or 400 images) and tested on a DrivingStereo sequence (6905 frames).

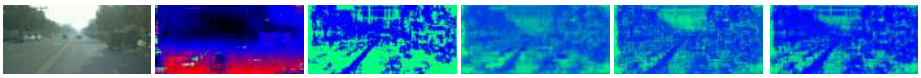

Fig. 4. Qualitative results on DrivingStereo. From left: reference image, disparity by Census-SGM, ConfNet trained with [58], [28], OTB and online-adapted OTB.

without retraining or adaptation, enabling to assess the generalization properties of each network/supervision configuration. We point out how the margin between learned and traditional measures is much smaller because of the domain shift. Nonetheless, in many cases the performance is still in favor of learned approaches, with some exceptions. We point out that networks trained with OTB self-supervision always outperform SELF and WILD, except for ConfNet with Census-SGM on ETH3D. Moreover, networks trained with OTB generalize better than their fully supervised counterparts in some cases, mostly on the ETH3D dataset (e.g., CCNN with all algorithms, ConfNet with MCCNN-fst-SGM and LGC with both SGM methods). This supports the better generalization achieved when training with OTB. Finally, Fig. 3 shows qualitative examples of this test.

4.4 Self-adapting in-the-wild

Finally, we conduct experiments aimed at assessing how effective our strategy is for self-adaptation of a confidence measure in unseen environments. Purposely, we simulate deployment in an autonomous driving scenario, selecting a sequence from the DrivingStereo dataset [60]. We use sequence 2018-10-25-07-37, containing 6905 stereo pairs acquired in unconstrained (i.e., dynamic) environment. For this evaluation, we choose Census-SGM, MADNet and GANet. The former because it represents the preferred choice for hardware implementation on custom stereo cameras [1, 8, 45, 16, 25, 42, 41]. The remaining two because well representing modern end-to-end CNNs that are fast (MADNet) or yield state-of-the-art accuracy (GANet). For confidence networks, we select ConfNet since it yielded excellent performance in the previous experiments, especially with accurate algorithms, and well-suited for online adaptation.

In this experiment, we assume to have pre-trained versions of ConfNet with the different self-supervision paradigms, again on KITTI 2012. For OTB, we use $[\mathcal{T}^p, \mathcal{A}^p, \mathcal{U}^p, \mathcal{T}^q]$ for SGM, $[\mathcal{T}^p, \mathcal{A}^p, \mathcal{U}^p, \mathcal{T}^q, \mathcal{A}^q, \mathcal{U}^q]$ offline and $[\mathcal{T}^p, \mathcal{A}^p, \mathcal{U}^p, \mathcal{T}^q]$ online for MADNet, $[\mathcal{T}^p, \mathcal{A}^p, \mathcal{U}^p, \mathcal{T}^q]$ for GANet. When performing online adaptation (**online** entry), for each stereo pair the confidence is estimated and evaluated **before** loss computation (thus, supervision only acts on the upcoming

Fig. 5. Qualitative results with Apple iPhone XS. We show two examples of reference image and disparity map acquired with the iPhone XS, followed by estimated confidence map after few iterations of on-the-fly learning.

frames as in [56]). This way, ConfNet runs at 0.08 seconds (12 FPS) against the 0.02 (50 FPS) without adaptation on Titan Xp. Table 4 collects the outcome of this evaluation. We point out that WILD can not be deployed for MADNet and GANet since a meaningful cost volume is not available for the former or cannot used straightforwardly for the latter. On the other hand, SELF would require $(\mathcal{D}_L, \mathcal{D}_R)$ for supervision, while MADNet and GANet provide only the former. Assuming networks as a gray-box, we get rid of this issue at training time obtaining $\mathcal{D}_R$ as shown in Eq. 1. Concerning SGM, OTB performs in between WILD and SELF. Nevertheless, keeping continuous adaptation active on the whole sequence makes it outperform both by a good margin. Concerning MADNet, SELF results more effective than OTB. Again, performing online adaptation makes OTB the best solution in this case as well. Finally, concerning GANet, SELF and OTB result equivalent, with online adaptation resulting crucial for this latter to achieve the best results. Anyway, such improvement saturates with the performance of the reprojection error, shown in column 1. To conclude, Fig. 4 shows qualitative examples for the SGM algorithm.

On-the-fly learning with black-box sensors. Finally we report, as qualitative results, the outcome obtained by learning on-the-fly a confidence measure on disparity map sourced by an Apple iPhone XS, without any pre-training. Fig. 5 shows examples of acquired disparity and estimated confidence maps by ConfNet adapted online. In particular, about 100 frames are sufficient to learn how to detect gross errors like on turtle's shell.

5 Conclusion

In this paper, we have introduced a novel self-supervised paradigm aimed at learning from scratch a confidence measure for stereo. We leverage few, principled cues from the input stereo pair and the estimated disparity in order to source supervision signals in place of disparity ground truth labels. Being such cues available during deployment in-the-wild, our solution is suited for continuous online adaptation on any black-box framework. Experimental results proved that our strategy is equivalent or superior to existing self-supervised approaches and, conversely to them, allow to further improvements during deployment by leveraging the online self-adaptation process.

References

1. Banz, C., Hesselbarth, S., Flatt, H., Blume, H., Pirsch, P.: Real-time stereo vision system using semi-global matching disparity estimation: Architecture and fpga-implementation. In: ICSAMOS. pp. 93–101 (2010)
2. Casser, V., Pirk, S., Mahjourian, R., Angelova, A.: Depth prediction without the sensors: Leveraging structure for unsupervised learning from monocular videos. In: Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19) (2019)
3. Chang, J.R., Chen, Y.S.: Pyramid stereo matching network. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
4. Chen, Z., Sun, X., Wang, L., Yu, Y., Huang, C.: A deep visual correspondence embedding model for stereo matching costs. In: The IEEE International Conference on Computer Vision (ICCV) (December 2015)
5. Di Stefano, L., Marchionni, M., Mattoccia, S.: A fast area-based stereo matching algorithm. Image and Vision Computing **22**(12), 983–1005 (2004)
6. Duggal, S., Wang, S., Ma, W.C., Hu, R., Urtasun, R.: Deeppruner: Learning efficient stereo matching via differentiable patchmatch. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4384–4393 (2019)
7. Fu, Z., Fard, M.A.: Learning confidence measures by multi-modal convolutional neural networks. In: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 1321–1330. IEEE (2018)
8. Gehrig, S.K., Eberli, F., Meyer, T.: A real-time low-power stereo vision engine using semi-global matching. In: ICVS. pp. 134–143 (2009)
9. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? The KITTI vision benchmark suite. In: CVPR (2012)
10. Godard, C., Mac Aodha, O., Brostow, G.J.: Unsupervised monocular depth estimation with left-right consistency. In: CVPR (2017)
11. Godard, C., Mac Aodha, O., Brostow, G.J.: Digging into self-supervised monocular depth estimation. In: ICCV (2019)
12. Gul, M.S.K., Bätz, M., Keinert, J.: Pixel-wise confidences for stereo disparities using recurrent neural networks. In: BMVC (2019)
13. Haeusler, R., Nair, R., Kondermann, D.: Ensemble learning for confidence measures in stereo vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 305–312 (2013)
14. Hirschmuller, H.: Accurate and efficient stereo processing by semi-global matching and mutual information. In: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. vol. 2, pp. 807–814. IEEE (2005)
15. Hirschmüller, H., Innocent, P.R., Garibaldi, J.: Real-time correlation-based stereo vision with reduced border errors. International Journal of Computer Vision **47**(1-3), 229–246 (2002)
16. Honegger, D., Oleynikova, H., Pollefeys, M.: Real-time and low latency embedded computer vision hardware based on a combination of fpga and mobile cpu. In: IROS (2014)
17. Hu, X., Mordohai, P.: A quantitative evaluation of confidence measures for stereo vision **34**(11), 2121–2133 (2012)
18. Ilg, E., Saikia, T., Keuper, M., Brox, T.: Occlusions, motion and depth boundaries with a generic network for optical flow, disparity, or scene flow estimation. In: 15th European Conference on Computer Vision (ECCV) (2018)

19. Kendall, A., Martirosyan, H., Dasgupta, S., Henry, P., Kennedy, R., Bachrach, A., Bry, A.: End-to-end learning of geometry and context for deep stereo regression. In: The IEEE International Conference on Computer Vision (ICCV) (Oct 2017)
20. Kim, S., Kim, S., Min, D., Sohn, K.: Laf-net: Locally adaptive fusion networks for stereo confidence estimation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
21. Kim, S., Min, D., Kim, S., Sohn, K.: Feature augmentation for learning confidence measure in stereo matching. IEEE Transactions on Image Processing **26**(12), 6019–6033 (2017)
22. Liang, Z., Feng, Y., Guo, Y., Liu, H., Chen, W., Qiao, L., Zhou, L., Zhang, J.: Learning for disparity estimation through feature constancy. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
23. Luo, W., Schwing, A.G., Urtasun, R.: Efficient deep learning for stereo matching. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5695–5703 (2016)
24. Marin, G., Zanuttigh, P., Mattoccia, S.: Reliable fusion of tof and stereo depth driven by confidence measures. In: European Conference on Computer Vision. pp. 386–401. Springer (2016)
25. Mattoccia, S., Poggi, M.: A passive rgbd sensor for accurate and real-time depth sensing self-contained into an fpga. In: 9th ICDSC (2015)
26. Mayer, N., Ilg, E., Hausser, P., Fischer, P., Cremers, D., Dosovitskiy, A., Brox, T.: 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) (June 2016)
27. Menze, M., Geiger, A.: Object scene flow for autonomous vehicles. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
28. Mostegel, C., Rumpler, M., Fraundorfer, F., Bischof, H.: Using self-contradiction to learn confidence measures in stereo vision. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2016)
29. Park, M.G., Yoon, K.J.: Leveraging stereo matching with learning-based confidence measures. pp. 101–109 (2015)
30. Park, M.G., Yoon, K.J.: Learning and selecting confidence measures for robust stereo matching. IEEE transactions on pattern analysis and machine intelligence **41**(6), 1397–1411 (2018)
31. Poggi, M., Agresti, G., Tosi, F., Zanuttigh, P., Mattoccia, S.: Confidence estimation for tof and stereo sensors and its application to depth data fusion. IEEE Sensors Journal (2019)
32. Poggi, M., Aleotti, F., Tosi, F., Mattoccia, S.: Towards real-time unsupervised monocular depth estimation on CPU. In: IEEE/JRS Conference on Intelligent Robots and Systems (IROS) (2018)
33. Poggi, M., Mattoccia, S.: Deep stereo fusion: combining multiple disparity hypotheses with deep-learning. In: 2016 Fourth International Conference on 3D Vision (3DV). pp. 138–147. IEEE (2016)
34. Poggi, M., Mattoccia, S.: Learning a general-purpose confidence measure based on o (1) features and a smarter aggregation strategy for semi global matching. pp. 509–518. IEEE (2016)
35. Poggi, M., Mattoccia, S.: Learning from scratch a confidence measure. In: BMVC (2016)
36. Poggi, M., Mattoccia, S.: Learning to predict stereo reliability enforcing local consistency of confidence maps. pp. 2452–2461 (2017)

37. Poggi, M., Pallotti, D., Tosi, F., Mattoccia, S.: Guided stereo matching. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
38. Poggi, M., Tosi, F., Mattoccia, S.: Even more confident predictions with deep machine-learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 76–84 (2017)
39. Poggi, M., Tosi, F., Mattoccia, S.: Quantitative evaluation of confidence measures in a machine learning world. pp. 5228–5237 (2017)
40. Poggi, M., Tosi, F., Mattoccia, S.: Learning a confidence measure in the disparity domain from o (1) features. Computer Vision and Image Understanding **193**, 102905 (2020)
41. Rahnama, O., Cavallari, T., Golodetz, S., Tonioni, A., Joy, T., Di Stefano, L., Walker, S., Torr, P.H.: Real-time highly accurate dense depth on a power budget using an fpga-cpu hybrid soc. IEEE Transactions on Circuits and Systems II: Express Briefs **66**(5), 773–777 (2019)
42. Rahnama, O., Cavalleri, T., Golodetz, S., Walker, S., Torr, P.: R3sgm: Real-time raster-respecting semi-global matching for power-constrained systems. In: 2018 International Conference on Field-Programmable Technology (FPT). pp. 102–109. IEEE (2018)
43. Scharstein, D., Hirschmüller, H., Kitajima, Y., Krathwohl, G., Nešić, N., Wang, X., Westling, P.: High-resolution stereo datasets with subpixel-accurate ground truth. In: German conference on pattern recognition. pp. 31–42. Springer (2014)
44. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. International journal of computer vision **47**(1-3), 7–42 (2002)
45. Schmid, K., Hirschmuller, H.: Stereo vision and imu based real-time ego-motion and depth image computation on a handheld device. In: ICRA (2013)
46. Schonberger, J.L., Sinha, S.N., Pollefeys, M.: Learning to fuse proposals from multiple scanline optimizations in semi-global matching. pp. 739–755 (2018)
47. Schops, T., Schonberger, J.L., Galliani, S., Sattler, T., Schindler, K., Pollefeys, M., Geiger, A.: A multi-view stereo benchmark with high-resolution images and multi-camera videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3260–3269 (2017)
48. Seki, A., Pollefeys, M.: Patch based confidence prediction for dense disparity map. In: BMVC. vol. 2, p. 4 (2016)
49. Song, X., Zhao, X., Hu, H., Fang, L.: Edgestereo: A context integrated residual pyramid network for stereo matching. In: 14th Asian Conference on Computer Vision (ACCV) (2018)
50. Spyropoulos, A., Komodakis, N., Mordohai, P.: Learning to detect ground control points for improving the accuracy of stereo matching. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1621–1628 (2014)
51. Spyropoulos, A., Mordohai, P.: Ensemble classifier for combining stereo matching algorithms. In: 2015 International Conference on 3D Vision. pp. 73–81. IEEE (2015)
52. Spyropoulos, A., Mordohai, P.: Correctness prediction, accuracy improvement and generalization of stereo matching using supervised learning. International Journal of Computer Vision **118**(3), 300–318 (2016)
53. Tonioni, A., Poggi, M., Mattoccia, S., Di Stefano, L.: Unsupervised adaptation for deep stereo. In: The IEEE International Conference on Computer Vision (ICCV) (Oct 2017)

54. Tonioni, A., Poggi, M., Mattoccia, S., Di Stefano, L.: Unsupervised domain adaptation for depth prediction from images. IEEE Transactions on Pattern Analysis and Machine Intelligence (2019)
55. Tonioni, A., Rahnama, O., Joy, T., Di Stefano, L., Thalaiyasingam, A., Torr, P.: Learning to adapt for stereo. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
56. Tonioni, A., Tosi, F., Poggi, M., Mattoccia, S., Di Stefano, L.: Real-time self-adaptive deep stereo (June 2019)
57. Tosi, F., Poggi, M., Benincasa, A., Mattoccia, S.: Beyond local reasoning for stereo confidence estimation with deep learning. pp. 319–334 (2018)
58. Tosi, F., Poggi, M., Tonioni, A., Di Stefano, L., Mattoccia, S.: Learning confidence measures in the wild. In: BMVC (Sept 2017)
59. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)
60. Yang, G., Song, X., Huang, C., Deng, Z., Shi, J., Zhou, B.: Drivingstereo: A large-scale dataset for stereo matching in autonomous driving scenarios. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
61. Yang, G., Zhao, H., Shi, J., Deng, Z., Jia, J.: Segstereo: Exploiting semantic information for disparity estimation. In: 15th European Conference on Computer Vision (ECCV) (2018)
62. Yin, Z., Darrell, T., Yu, F.: Hierarchical discrete distribution decomposition for match density estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6044–6053 (2019)
63. Zbontar, J., LeCun, Y.: Stereo matching by training a convolutional neural network to compare image patches. Journal of Machine Learning Research **17**(1-32), 2 (2016)
64. Zhang, F., Prisacariu, V., Yang, R., Torr, P.H.: Ga-net: Guided aggregation net for end-to-end stereo matching. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 185–194 (2019)
65. Zhang, K., Lu, J., Lafruit, G.: Cross-based local stereo matching using orthogonal integral images. IEEE transactions on circuits and systems for video technology **19**(7), 1073–1079 (2009)
66. Zhang, Z., Cui, Z., Xu, C., Jie, Z., Li, X., Yang, J.: Joint task-recursive learning for semantic segmentation and depth estimation. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 235–251 (2018)
67. Zhong, Y., Li, H., Dai, Y.: Self-supervised learning for stereo matching with self-improving ability. arXiv preprint arXiv:1709.00930 (2017)
68. Zhong, Y., Li, H., Dai, Y.: Open-world stereo video matching with deep rnn. In: ECCV (2018)