Good cues to learn from scratch a confidence measure for passive depth sensors

Matteo Poggi, *Member, IEEE,* Fabio Tosi, *Student Member, IEEE,* and Stefano Mattoccia, *Member, IEEE*

***Abstract*—As reported in the stereo literature, confidence estimation represents a powerful cue to detect outliers as well as to improve depth accuracy. Purposely, we proposed a strategy enabling us to achieve state-of-the-art results by learning a confidence measure in the disparity domain only with a CNN. Since this method does not require the cost volume, it is very appealing because potentially suited for any depth-sensing technologies, including, for instance, those based on deep networks. By following this intuition, in this paper, we deeply investigate the performance of confidence estimation methods, known in the literature and new ones proposed in this paper, neglecting the use of the cost volume. Specifically, we estimate from scratch confidence measures feeding deep networks with raw depth estimates and optionally images and assess their performance deploying three datasets and three stereo algorithms. We also investigate, for the first time, their performance with disparity maps inferred by deep stereo end-to-end architectures. Moreover, we move beyond the stereo matching context, estimating confidence from depth maps generated by a monocular network. Our extensive experiments with different architectures highlight that inferring confidence prediction from the raw reference disparity only, as proposed in our previous work, is not only the most versatile solution but also the most effective one in most cases.**

I. Introduction

The availability of accurate 3D data is of paramount importance for a large number of high-level tasks in computer vision and, purposely, some active sensing technologies exist. Some of them are particularly effective for outdoor environments (e.g., LiDAR) while others for indoor (e.g., devices based on light pattern projection or Time-Of-Flight technology). However, regardless of the technology deployed, they require to perturb the sensed area with signals leading to poor performance, for instance, with reflective or absorbing materials.

On the other hand, passive depth sensing techniques have the potential to overcome all these issues by inferring depth with standard imaging devices. Although various approaches exist, stereo represents the most popular and effective technique for this purpose. Despite the high accuracy achieved by state-of-the-art stereo algorithms, this technology suffers from some intrinsic limitations. For example, occluded areas, low textured and ambiguous regions such as reflective surfaces, are challenging and thus prone to errors. Therefore, by reliably detecting wrong depth measurements, one can remove or replace outliers preventing possible failures of high-level applications.

Confidence measures [1], [2] proved to be very effective to detect wrong measurements as well as to improve the overall accuracy of stereo matching [3], [4], [5], [6], [7]. Moreover, their deployment was also beneficial for other purposes such as the self-supervised adaptation of deep networks [8], [9], self-supervised training of confidence measures [10], and sensor fusion [11], [12]. Recent works concerning confidence estimation highlighted that the disparity domain contains enough information to detect outliers effectively [13], [13], [14], [15], [7], enabling to accomplish this task even when the *cost volume* is not available. This fact occurs, for instance, when dealing with deep networks for stereo and monocular depth estimation or with off-the-shelf stereo cameras. Moreover, it is also worth pointing out that enabling accurate detection of outliers from depth data could be potentially useful for additional purposes too, for instance, for registration of data with a different modality [16].

In our previous work [13], we showed for the first time that a *Confidence Convolutional Neural Network* (CCNN) could be trained for state-of-the-art confidence estimation from a single disparity map. Currently, such a network also represents the basic building block of top-performing methods, processing either the disparity map [15], [17] or the cost volume [18].

Following this path, in this paper, we deeply evaluate which features traditionally available from any depth camera, i.e. disparity map(s) and the RGB image(s), are relevant to estimate confidence when fed to a deep network trained for this purpose. To assess the importance of such features, we carry out an exhaustive evaluation with three standard datasets and three popular stereo algorithms [2]. Moreover, since end-to-end stereo architectures represent the state-of-the-art for stereo, we show that the considered methods can infer, even in this case, a meaningful confidence estimation whereas other known techniques based on cost volume processing could not. Finally, we move beyond stereo matching and evaluate the considered confidence estimation methods with the maps generated by deep networks for monocular depth perception.

Our evaluation highlights that CCNN is not only the most versatile method, being suited for any depth sensing device, but also, in most cases, the most effective confidence estimation approach for depth data. Nonetheless, in some specific circumstances, adding additional cues such as RGB image(s) and the target disparity map, despite this latter cue is not always available, yields slightly better accuracy.

II. Related work

Confidence measures for stereo. Confidence measures were first extensively reviewed and categorized by [1], and more recently by [2] considering learning-based approaches. Both works emphasize how different cues can be taken into account to formulate a confidence score, such as: matching

M. Poggi, F. Tosi and S. Mattoccia are with the University of Bologna.

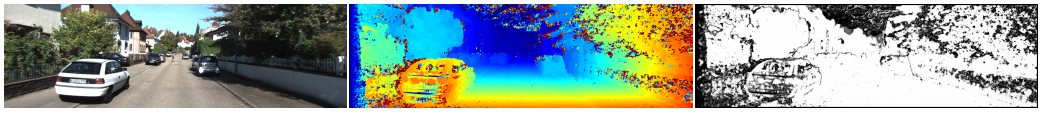

Fig. 1. **Confidence estimation in the disparity domain.** By visually inspecting reference image (left) and corresponding disparity map (center) computed by a stereo algorithm, outliers can be easily identified. On the same principle, a CNN can predict a reliable confidence map (right) processing the same cues.

cost, local or global property of the cost curve, left-right consistency and others. Single confidence measures can be effectively combined with other hand-crafted features and fed to a classifier (usually, a random forest) to learn a more accurate confidence score [19]. Some works [20], [4], [6], [7] adopted this rationale deploying different features. Moreover, leveraging the learned confidence estimation, these methods enabled improvements to stereo accuracy.

Eventually, confidence estimation was tackled exploiting CNNs. Specifically, in [13] we proposed to learn from scratch a confidence measure training the CCNN network on samples extracted from the raw reference disparity map only, while [5] processing hand-crafted features extracted from the reference and target disparity maps for their *Patch Based Confidence Prediction* (PBCP) approach. In [14] the additional contextual information from the reference frame is exploited. However, this method requires a much larger training set compared to any other method discussed so far because of the increased variety of data occurring in the image domain. In [21] a comparison between random forest and CNN processing the same features is reported. Differently, arguing local consistency of confidence maps, in [17], [22] deep networks were trained to improve the overall accuracy of an input confidence map. Other effective strategies consist in combining local and global cues from both image and disparity domains as proposed by [15] or adding features computed from the cost volume as shown by [18]. An evaluation of confidence measures suited for embedded devices was proposed in [23]. Finally, [10] and [24] proposed two strategies to train confidence measures without ground-truth labels.

Applications of confidence measures. Confidence estimation has been recently deployed beyond their original outlier detection scope. Mainly, it has been used to improve stereo accuracy by detecting ground control points [20], to smooth the cost volume [4], to improve Semi Global Matching [25] (SGM) by better combining scanline optimizations [6], [7], [3] or dynamicaly adjusting penalties [5].

Other applications of confidence prediction concern fusion of depth sensors with different technology [11], [12], disparity fusion [26], [27] self-supervised adaptation of deep stereo models [8], [9] and self-supervised learning of confidence measures from stereo pairs [10].

Stereo matching. Inferring depth from a couple of synchronized images represents one of the most popular techniques in computer vision. Conventional stereo algorithms are classified [28] in local and global, according to the subset of steps performed, namely i) cost computation ii) cost aggregation iii) disparity optimization and iv) refinement. Common to all strategies is matching cost computation, relying on the simple Sum of Absolute Differences (SAD) or more robust metrics [29], with census transform [30] often being the preferred choice. Among traditional algorithms, SGM [25] represents a good trade-off between speed and accuracy. Nonetheless, the advent of large and challenging datasets with available ground truth depth labels, such as KITTI 2012 [31], KITTI 2015 [32] and Middlebury 2014 [33] highlighted that, despite traditional algorithms have excellent performance on controlled environments, they are still far from optimal results when used in real applications such as autonomous driving.

Deep learning for stereo. The work by Zbontar and LeCun [34] represented the very first attempt to use deep learning to tackle stereo matching with a *Matching Cost CNN*, namely MC-CNN. Compared to conventional matching strategies [29], the outcome is a much more effective cost function according to the evaluation on KITTI and Middlebury datasets. Other works followed this strategy for matching cost computation: [35] designed more robust representations while [36], [37] proposed faster architectures. Later works proved that dense disparity estimation could be tackled in an end-to-end fashion with deep learning models trained to infer per-pixel values directly. A seminal work in this field was proposed by [38] introducing DispNetC, an encoder-decoder architecture implicitly solving stereo correspondence from scratch. Latest works achieved state-of-the-art results on KITTI datasets by designing end-to-end architectures, with modules explicitly dealing with the standard phases [28] of a conventional stereo pipeline. Notable examples are GC-Net [39], iResNet [40], PSMNet [41], GA-Net [42] and GWC-Net [43]. To deal with out-of-distribution data, some recent works proposed self-adapting frameworks [44], [45] and guided deep networks [46], enabling to take advantage of sparse reliable depth measurements.

III. Learning from scratch confidence measures

A recent trend concerning confidence estimation proves that it can be reliably inferred in the disparity domain. The primary rationale behind this strategy is that, by visual inspection, several outliers can be easily spotted from the disparity assignments of the neighboring pixels, as evident from Fig. 1.

Purposely, in this paper, we thoroughly investigate strategies to learn from scratch a confidence measure with deep learning by relying on visual cues only and neglecting the use of the cost volume, seldom available outside of the conventional stereo context. For instance, nowadays, the outlined circumstance frequently occurs in very relevant cases, such as when dealing with custom stereo cameras, deep stereo [38] and monocular [47] depth estimation networks.

A. Input cues

In order to avoid the need for any intermediate representation, such as the cost volume, we leverage only on the visual

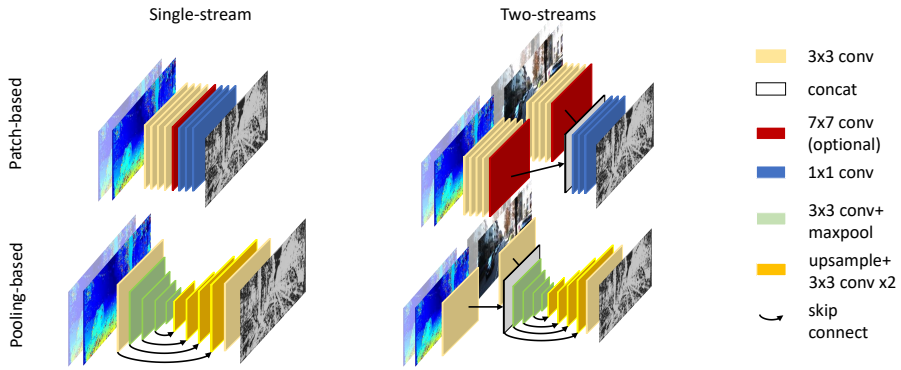

Fig. 2. **Confidence estimation networks.**. On the left, the single-stream models processing only disparity cues. On the right, two-streams models processing both RGB image and disparity cue separately. On top, patch-based networks, on bottom pooling-based. 7×7 convolutional layer is used only in case of 15×15 receptive field.

cues potentially available with a stereo or monocular setup. Specifically, in the stereo case, given the left and right images I_L and I_R and assuming the former as the reference one, we can always obtain a disparity map for the reference view and, possibly, a map for the target view, respectively D_L and D_R. Nonetheless, it is worth observing that D_R might be not available in some cases, for instance, when dealing with off-the-shelf stereo cameras (e.g., the Intel RealSense stereo camera).

Thus, in this paper, we consider an exhaustive combination of such input cues including configurations already evaluated and some never explored before. We compare with the same CNN baseline network different combinations of input features enabling to highlight which cues are truly useful for confidence estimation:

Reference disparity map D_L: the disparity map aligned with the input reference frame (typically, the left image). In [13] this cue proved to be sufficient to learn a confidence measure.

Reference image I_L: RGB input reference frame. [14] extended CCNN to account for this additional cue.

Warped target disparity map D'_R: obtained by warping target disparity map D_R into the left reference frame according to D_L. The absolute difference between D'_R and D_L encodes the *left-right difference* exploited by [5] with PBCP. Purposely, pixel at coordinates (x, y) is sampled at $(x - D_L(x, y), y)$ from D_R as $D'_R(x, y) = D_R(x - D_L(x, y), y)$. This configuration, referred to as *fast* in [5], is suited for processing in fully convolutional manner and thus compatible with both network architectures adopted in our experiments and detailed in the reminder, conversely to other configurations in [5] that require independent processing of each single patch.

Warped target image I'_R: image obtained by warping I_R into the left reference frame according to D_L. To the best of our knowledge, this cue has never been considered before for confidence prediction. Pixel at coordinates (x, y) is sampled at $(x - D_L(x, y), y)$ from I_R as $I'_R(x, y) = I_R(x - D_L(x, y), y)$.

By designing fully-convolutional architectures, in a single forward pass, we can estimate confidence for all image pixels.

B. Network architectures and configurations

In the literature, different CNN architectures have been deployed for confidence estimation. In the remainder, to adequately assess the contribution given by the different input cues to the final confidence estimation, we focus on two main categories:

Patch-based [13], [5], [14], made by convolutional layers only. Spatial resolution is reduced by convolving over valid pixels (i.e., without padding). For instance, 3×3 convolutions reduce the resolution by 2 pixels on each dimension, thus processing a single output value from a 3×3 patch.

Pooling-based [15], decimating the resolution by means of pooling operations. The original resolution is restored for the output through deconvolutions or upsampling operations.

For both, we deploy a baseline architecture regarding the number of convolutional layers, channels, activation layers and input dimensions. Precisely, we deploy the architecture from [13] for patch-based family and ConfNet [15] for pooling-based. In this latter case, we replace deconvolutions with nearest neighbour upsampling followed by convolutions to improve accuracy. The final outputs are normalized in [0, 1] by a sigmoid layer.

Figure 2 depicts the architectures outlined so far, respectively patch-based (top) and pooling-based (bottom). Regarding patch-based models, we consider two variants with different receptive field as proposed in the literature: 9×9 [13] and 15×15 [5]. For this latter, a 7×7 layer (red in figure) is added to reduce features dimensions to 1×1 as well[1].

We consider different combinations of the input cues mentioned above, leading overall to the following six network configurations:

- **CCNN** – our *Confidence Convolutional Neural Network* processing D_L only [13]
- **LF** – *Late Fusion* of D_L and I_L [14]
- **PBCP** – *Patch Based Confidence Prediction* from D_L and D'_R [5]

[1]While 9×9 patches are reduced by 4 layers to a single pixel, 15×15 patches are reduced to 7×7 as in [5]. The 7×7 layer reduces these latter to a single pixel as well

- **LF-PBCP** – a model mixing information from D_L, D'_R and the reference image I_L
- **LRLF** – a late fusion model combining I_L, I'_R with reference disparity map D_L
- **LRLF-PBCP** – a network processing all the information available from a stereo setup: D_L, D'_R, I_L and I'_R

CCNN and PBCP rely on a single-stream architecture while the others on two streams. The two resulting variants, for each family, are depicted respectively at the left and right of Figure 2. The two streams are combined using concatenation (white layers). While CCNN, LF and PBCP are known in the literature, LF-PBCP, LRLF and LRLF-PBCP are new proposals. Therefore, given the 6 configurations, the 2 patch-based and the pooling-based models, a total of 18 networks will be evaluated.

IV. Experimental results

In this section, we report an exhaustive evaluation concerning the models introduced above.

A. Algorithms and networks for depth from passive sensors

We consider in our evaluation traditional and learning-based stereo algorithms, an end-to-end model to infer depth from stereo pairs and a monocular depth estimation network.

AD-CENSUS [30], obtained by applying the census transform with window size 5×5 and computing the difference between left and right transformed images according to the Hamming distance. A 5×5 box filtering operation is applied before the *winner takes all* (WTA) strategy. This algorithm is generally considered as the baseline when evaluating confidence measure [2].

MC-CNN [37], CNN based matching cost processing 9×9 or 15×15 patches (on KITTI and Middlebury, respectively). As in previous studies [2], in our evaluation we consider MC-CNN-fst, trained on each dataset by the same authors. From MC-CNN-fst matching costs, disparity selection is carried out according to the WTA strategy.

SGM [25], using eight scanline optimizations and cost volume obtained by normalized AD-CENSUS score. We tuned penalties to be P1=0.03, P2=3.

DispNetC [38], an encoder-decoder CNN inferring out-of-the-box a disparity map given a stereo pair. Specifically, we use the weights made available by the same authors. The output of this network is the disparity map computed according to the reference image only. Therefore, with such an algorithm, the available input clues fit only with CCNN, LF and LRLF.

Monodepth [47], an encoder-decoder model inferring inverse depth maps (i.e., disparity) from a single input image. The network is trained in a self-supervised manner using image reconstruction losses on frames acquired by a stereo rig. In our experiments, we consider the VGG model trained by the same authors. Concerning this method, only CCNN and LF are compatible with the available input clues.

B. Implementation details and training procedure

All models have been implemented using the TensorFlow framework. Patch-based models have been trained on batches of 128 image patches, while and pooling-based models on batches of 4 crops of size 256×512, both with Binary Cross Entropy (BCE) loss between estimated confidence c_i and ground truth confidence label y_i, for central pixel i in each patch in the former case or for all the pixels in each crop in the latter. Labels y_i are set to 0 if, in the case of stereo algorithms, the difference between estimated and ground-truth disparity is higher than a threshold τ and 1 otherwise. For Monodepth, we follow a slightly different protocol, described in detail in Sec. IV-E.

We used Stochastic Gradient Descent (SGD) as the optimizer and a learning rate of 3×10^{-3}. Training samples have been generated inferring disparity maps for each of the five considered depth sensing methods on the KITTI 2012 dataset. In particular, we sampled two different training splits out of the total 200 stereo pairs with ground truth depth labels available:

KITTI-small, made of the first 20 frames [2], providing about 2.7 million pixels with available ground truth labels.

KITTI-large, made of the first 94 frames [14]. This configuration yields about 8 million depth samples.

In order to assess the performance of confidence prediction across different domains, we train on both splits and test on remaining samples from KITTI 2012, as well as on KITTI 2015 and Middlebury without re-training the networks [2] to highlight how each input feature or their combinations are robust to domain shift. Indeed, since all the networks have been designed starting from the same baseline structure, such evaluation will be able to assess the impact of each input cue. Given the six combinations of input cues, the two patch-based and the pooling-based models and the two training portions described so far we trained: 12 models for the Monodepth algorithm, 18 for DispNetC and 36 for each stereo algorithms. Overall, we trained 138 networks in about one week with an NVIDIA Titan Xp GPU. Concerning runtime, Table I reports the time required on the same GPU to estimate a confidence map at KITTI resolution (about 375×1242), showing almost equivalent runtime for the two patch-based models. Pooling-based models are much faster, thanks to the reduction of spatial resolution, but less accurate as we will see through the evaluation reported next.

TABLE I
Runtime on KITTI images (375×1242) on a Titan Xp GPU.

Architecture	Single-Stream			Two-Streams		
Variant	9×9	15×15	Pool	9×9	15×15	Pool
Runtime	0.07 s	0.10s	0.02	0.09s	0.13s	0.03

C. Evaluation protocol

To quantitatively measure the effectiveness of confidence prediction, we use the standard protocol adopted in this field [1], [2]. To this aim, given a disparity map, we sort all pixels according to their confidence scores in descending order. Then, a defined sampling interval is used to iteratively extract a fixed number of samples from the sorted set of pixels and compute the percentage of outliers after each extraction. The outcome of this process is a discretized curve from which we compute the Area Under the Curve (AUC). An optimal confidence measure

TABLE II
AVERAGE AUC MARGIN (%) ON DISPARITY MAPS BY DIFFERENT STEREO METHODS.

	KITTI 2012						KITTI 2015						Middlebury 2014					
	Patch (9x9)		Patch (15x15)		Pool		Patch (9x9)		Patch (15x15)		Pool		Patch (9x9)		Patch (15x15)		Pool	
	small	large	small	large	small	large	small	large	small	large	small	large	small	large	small	large	small	large
CCNN	12.84	13.14	17.71	13.33	20.52	19.03	15.97	16.76	21.97	17.21	28.88	26.39	25.03	22.36	30.03	24.92	43.27	37.82
PBCP	16.03	14.11	22.21	17	20.62	24.64	22.2	20.16	29.11	23.56	28.88	33.18	32.26	22.47	32.7	30.92	40.6	40.49
LF	13.78	14.88	21.93	14.88	19.96	18.65	17.33	18.01	27.97	20.16	27.07	25.71	30.59	26.03	40.16	31.37	39.49	37.26
LF-PBCP	15.09	13.62	23.62	16.43	22.68	20.1	21.74	19.48	32.16	23.33	30.58	27.29	32.93	28.92	43.6	42.83	47.27	44.83
LRLF	14.06	14.2	20.15	13.72	21.74	25.6	17.1	18.35	25.59	18.35	28.09	36.24	26.81	27.7	41.71	28.59	45.38	45.94
LRLF-PBCP	15.93	15.17	24.46	16.62	20.43	30.53	22.76	21.52	34.43	23.56	27.18	37.49	29.25	34.15	51.61	33.93	39.27	60.18

(a) CENSUS algorithm

	KITTI 2012						KITTI 2015						Middlebury 2014					
	Patch (9x9)		Patch (15x15)		Pool		Patch (9x9)		Patch (15x15)		Pool		Patch (9x9)		Patch (15x15)		Pool	
	small	large	small	large	small	large	small	large	small	large	small	large	small	large	small	large	small	large
CCNN	26.41	22.32	39.39	28.57	41.99	41.52	37.26	35.38	50.47	40.57	54.25	56.13	39.08	36.03	46.29	35.37	41.7	41.92
PBCP	29.44	22.77	42.42	28.57	35.93	33.04	42.45	37.26	53.3	41.98	50.94	46.7	40.83	37.34	49.78	34.93	40.39	37.77
LF	28.14	22.77	39.83	31.25	48.05	51.79	38.68	39.15	51.42	42.45	68.87	69.81	48.47	46.94	56.99	51.53	89.96	89.74
LF-PBCP	30.3	23.21	39.83	30.36	43.29	34.38	41.51	38.21	58.96	46.23	58.49	50.94	50.66	42.14	68.56	50.66	79.69	59.83
LRLF	29.44	23.21	35.93	35.27	44.16	41.96	43.87	36.32	50.47	49.06	61.32	59.43	40.17	42.79	48.91	50.44	83.19	79.26
LRLF-PBCP	29.87	23.66	38.1	29.02	36.36	36.16	43.4	38.21	54.72	42.92	51.89	52.36	46.07	55.46	62.88	46.51	63.1	63.76

(b) MC-CNN algorithm

	KITTI 2012						KITTI 2015						Middlebury 2014					
	Patch (9x9)		Patch (15x15)		Pool		Patch (9x9)		Patch (15x15)		Pool		Patch (9x9)		Patch (15x15)		Pool	
	small	large	small	large	small	large	small	large	small	large	small	large	small	large	small	large	small	large
CCNN	117.05	103.49	119.32	125.58	119.32	115.12	116.48	115.38	115.38	136.26	123.08	126.37	69.38	63.44	73.13	75.33	76.65	81.06
PBCP	115.91	105.81	110.23	102.33	128.41	110.47	117.58	117.58	114.29	119.78	135.16	116.48	72.25	68.28	77.31	68.5	85.02	75.99
LF	126.14	101.16	126.14	118.6	176.14	129.07	126.37	116.48	137.36	125.27	238.46	153.85	76.87	81.94	90.97	75.55	177.97	116.96
LF-PBCP	102.27	95.35	110.23	109.3	198.86	162.79	114.29	112.09	131.87	125.27	249.45	216.48	83.7	68.06	87.44	73.57	203.52	176.21
LRLF	129.55	106.98	246.59	146.51	234.09	109.3	124.18	126.37	260.44	148.35	258.24	143.96	83.04	88.55	135.68	74.89	231.28	129.07
LRLF-PBCP	112.5	96.51	128.41	111.63	303.41	163.95	116.48	113.19	140.66	124.18	327.47	198.9	79.3	75.99	80.4	70.26	281.94	163

(c) SGM algorithm

	KITTI 2012						KITTI 2015						Middlebury 2014					
	Patch (9x9)		Patch (15x15)		Pool		Patch (9x9)		Patch (15x15)		Pool		Patch (9x9)		Patch (15x15)		Pool	
	small	large	small	large	small	large	small	large	small	large	small	large	small	large	small	large	small	large
CCNN	549.4	473.26	655.42	636.05	1213.25	838.37	442.06	382.24	601.87	572.9	700	670.09	198.26	198.98	178.53	186.3	266.87	218.71
LF	628.92	515.12	603.61	702.33	706.02	605.81	502.8	412.15	478.5	619.63	506.54	584.11	215.13	201.64	207.16	202.04	236.3	219.63
LRLF	721.69	538.37	687.95	697.67	687.95	677.91	599.07	384.11	574.77	612.15	570.09	575.7	208.18	209.71	203.07	183.13	232.21	208.9

(d) DispNetC network

would allow sampling of all pixels with correct disparities first, thus resulting in the lowest AUC score. Thus, given a disparity map with a percentage ε of outliers, the optimal AUC value is obtained as follows [1]:

$$\text{AUC}_{opt} = \int_{1-\varepsilon}^{\varepsilon} \frac{p-(1-\varepsilon)}{p} dp = \varepsilon + (1-\varepsilon)\ln(1-\varepsilon) \quad (1)$$

Then, we measure the effectiveness of a confidence measure by computing the relative margin with respect of the optimal AUC score as $\frac{\text{AUC}-\text{AUC}_{opt}}{\text{AUC}_{opt}}$.

In our experiments, we sampled pixels at intervals of 5% of the overall amount and we label as outliers pixels with an absolute disparity error larger than 3 for KITTI datasets and 1 for Middlebury, this latter processed at quarter resolution. We refer to this amount of erroneous pixels as, respectively, *bad-3* and *bad-1* error rates. For training, we set τ to 3. As already pointed out, when dealing with a monocular network, we change the method to detect outliers, as will be explained in detail later.

Considering the vast amount of data collected, before reporting the outcome of our evaluation we describe in detail how to correctly parse the information provided. In particular, for each depth estimation method, we will report tables organized into three main blocks for the three datasets, respectively KITTI 2012, KITTI 2015 and Middlebury 2014 from left to right. Each block is divided into three groups of two columns regarding patch-based models (9×9 and 15×15) and the pooling-based model (Pool) as third. In each group, the two columns concern the KITTI-small and KITTI-large splits. Each row reports averaged AUC values for a specific combination of input cues. Thus, each score refers to a features configuration tested on a particular dataset after being trained on one of the two possible training splits. For each single dataset, we apply a heatmap to better distinguish top-performing configurations (in green) from those less effective (in red).

D. Evaluation with stereo algorithms

Table II reports results concerned with confidence estimation from disparity maps computed by the aforementioned stereo algorithms. We will now discuss on each one.

AD-CENSUS. Table II (a) reports results concerned AD-CENSUS algorithm [30]. Such a method achieves quite high error rate on the three datasets, respectively, about 39% bad-3, 35% bad-3 and 37% bad-1 on KITTI 2012, KITTI 2015 and Middlebury 2014. Nonetheless, its cost volume is often deployed inside well-known pipelines such as SGM [25]. Thus, inferring an effective confidence scoring in this case allows for deployment of techniques such as [20], [4], [5].

Concerning patch-based methods, we can notice from the table how the reference disparity map alone contains enough information to detect outliers with this stereo algorithm nearly

TABLE III
AVERAGE AUC MARGIN (%) ON DISPARITY MAPS BY MONODEPTH.

	KITTI 2012						KITTI 2015					
	Patch (9x9)		Patch (15x15)		Pool		Patch (9x9)		Patch (15x15)		Pool	
	small	large	small	large	small	large	small	large	small	large	small	large
CCNN	299.12	267.57	769.91	927.93	1251.33	359.46	304.46	264.97	893.63	1027.39	1317.83	325.48
LF	361.06	270.27	488.5	654.95	446.02	554.95	347.77	320.38	645.22	648.41	382.8	499.36

optimally. Indeed, CCNN always achieves the lowest average AUC margin . The reason is that the network effectively learns to detect from this cue glaring error patterns, clearly visible in the disparity map, as well as high-frequency noise often present in regions supposed to be smooth. Adding other cues reduces confidence estimation capability slightly, probably because they are scarcely informative (e.g., the algorithm often fails where the original stereo images lack information, as in textureless regions). Processing 9×9 patches catches enough context with all datasets, while including more training samples from KITTI 2012 (large vs small splits) improves the results on the remaining of KITTI 2012 and on Middlebury. Nonetheless, this strategy yields slightly worse accurate confidence prediction on KITTI 2015.

Pooling-based models, although faster, typically perform worse than patch-based methods confirming the findings in [15], because of pooling operations losing high-frequency details. Nonetheless, additional cues, e.g. the LF configuration, enable to reduce the gap on all the datasets partially.

MC-CNN. Table II (b) reports the outcome of the evaluation with MC-CNN-fst. This algorithm almost halves the number of outliers compared to AD-CENSUS, leading respectively to about 17%, 15% and 26% error rates on KITTI 2012, KITTI 2015 and Middlebury. Nevertheless, its local nature yields error patterns similar to those observed with AD-CENSUS.

Therefore for patch-based models, in most cases, the D_L features alone still leads to the best overall performance. The most notable exception is on Middlebury; in fact, using a 15×15 receptive field and KITTI-large for training, the warped disparity D'_R processed by PBCP enables to achieve slightly better confidence prediction accuracy compared to CCNN. This outcome might be the consequence of the stereo method itself processing larger patches (i.e., 11×11) on Middlebury. Moreover, in this case the large split is always beneficial to obtain the best accuracy, while the size of the receptive fields impacts differently according to the dataset.

Again, pooling-based models perform consistently worse than patch-based approaches, although they benefit more of additional information and achieve the best results with PBCP.

SGM. Table II (c) collects results concerning SGM, with error rates of about 9%, 10% and 27%.

For patch-based models, we can notice that even with much more accurate disparity maps, CCNN is still effective. Nonetheless, in this case, it is no longer the best overall solution to detect outliers as in the previous two experiments. In particular, PBCP improves confidence prediction of patch-based models on KITTI 2012 for three out of four cases, while adding the left image to CCNN seems effective only for 15×15 models. On KITTI 2015, mixed results are achieved by the three variants. However, LF-PBCP yields optimal performance on both KITTI datasets when training on the large split with a 9×9 receptive field. On the other hand, adding the warped right images seems not effective at all. Conversely, for Middlebury 2014 we can notice the best overall results are achieved by CCNN trained on KITTI-large with receptive field 9×9. Although the disparity cue alone is not optimal on data similar to the training images, it achieves better generalization across datasets, proving to be more robust when deployed in totally unseen environments.

Concerning pooling-based methods, in this case, more input cues seem useful only when more training data are available, with mixed configurations outperforming CCNN, still not matching the performance of patch-based networks.

DispNetC. Table II (d) reports the outcomes of experiments on DispNetC. Although this deep stereo network achieves about 6% and 7% bad-3 error rate on KITTI 2012 and 2015, its performance dramatically drops with the more heterogeneous indoor scenes depicted in the Middlebury 2014 dataset, falling to more than 30% bad-1 score. In this latter case, although the produced disparity maps look visually consistent, depth prediction is often inaccurate and detecting outliers becomes of paramount importance, yet very challenging. Being no cost volume available in this case, visual cues are crucial for the purpose. Since it provides a single disparity map aligned with the reference image without processing any traditional cost volume, only configurations CCNN, LF and LRLF are suitable. First of all, we can notice how AUC margins are much higher compared to what observed on previous evaluations. This fact can be explained considering the very accurate results yielded by the DispNetC network on KITTI datasets and the erroneous, yet visually consistent maps obtained on Middlebury.

For patch-based models, CCNN achieves the best results in most cases. In general, on KITTI the best results are achieved by 9×9 models, while on Middlebury 2014 the 15×15 architectures are more effective. LF seldom yields better performance (only in 2 out of 12 cases) and never enables to achieve the best accuracy. A possible cause is the fact that disparity maps produced by end-to-end models are particularly consistent with the reference image shape, thus adding such cue as input to detect outliers does not add particular information for the purpose. Interestingly, on Middlebury the best performance are achieved by training on the small split using a receptive field of 15×15.

As for traditional algorithms, pooling-based models cannot compete with patch-based ones and processing D_L alone leads most times to the worst results.

E. Evaluation with monocular depth estimation network

Finally, we inquire about the effectiveness of confidence measures initially conceived for stereo when dealing with monocular depth estimation networks. As a representative method in this field, we choose Monodepth [47], trained in a self-supervised manner on stereo pairs, for historical reasons and reproducibility. Given its input and output cues, only CCNN and LF can be deployed with this network. Moreover, we also point out that the authors have trained Monodepth on KITTI raw sequences [48], and thus it performs quite well in similar environments. However, it performs poorly when deployed in entirely different environments, such as the indoor scenes depicted in Middlebury 2014. For this reason, we evaluate the performance of confidence prediction frameworks for Monodepth only on the KITTI 2012 and 2015 datasets.

Metrics. In contrast to stereo, monocular depth estimation is an ill-posed problem enabling to infer depth up to a scale factor. Thus, we change the criterion to label pixels as outliers following the methodology used to evaluate monocular depth estimation methods. Explicitly, we follow [49] and consider outliers all pixels having $\max(\frac{D_L}{G}, \frac{G}{D_L}) > 1.25$, being D_L estimated depth and G ground truth depth. The same criterion is used at training time. Accordingly, Monodepth produces respectively 10% and 16% outliers on KITTI 2012 and 2015.

Evaluation. Table III collects the results regarding this experiment. Similarly to the evaluation with DispNetC, we can notice larger AUC margins compared to traditional stereo algorithms. Nonetheless, we can notice how CCNN always achieves the best accuracy for outliers detection among all considered measures with a 9×9 receptive field. In particular, trained on a smaller amount of images, it achieves the best results on KITTI 2015, granting better generalization capability. On the other hand, using 94 pairs for training yields optimal results on KITTI 2012 but at the same time, reduces confidence estimation accuracy on KITTI 2015. Adding the left image to the 9×9 networks does not increase accuracy except on KITTI 2015 when training on the more extensive training set. By enlarging the receptive field, CCNN loses accuracy. Processing the left image attenuates this effect, but still does not vouch for the same performance obtained by processing only the inverse depth map with a 9×9 receptive field. Concerning pooling-based strategy, this time it outperforms 15×15 patch-based networks when trained on a broader training set, but still cannot compete with 9×9. Surprisingly, CCNN configurations perform better than LF when trained on more samples.

F. Qualitative analysis

Finally, we report qualitative examples to highlight both the different nature of noise in the estimated disparity/depth maps and the effectiveness of confidence measures at detecting outliers. Figure 3 shows an example from the KITTI 2015 dataset. It reports the disparity map, for stereo, and the inverse depth maps, for the monocular network, generated by some of the method considered in our evaluation and the corresponding best confidence map for each one. We can notice how, with different degrees of reliability, a low confidence score (black) generally corresponds to an erroneous depth estimation on the top map. Figure 4 collects results on the *Adirondack* stereo pair from Middlebury 2014. We point out that, for the reasons outlined before, the confidence scores are more likely to fail in this case, for instance on the disparity map inferred by DispNetC (right), where part of the armchair is missing and the corresponding confidence values are high instead.

V. Conclusions

In this paper, we studied the importance of input cues processed to infer confidence scores from depth data generated by passive depth sensors with a CNN. Considering the same baseline architectures, we extensively assessed the performance of six models processing different input cues with different stereo and mono depth sensing strategies, including learning-based approaches.

Our in-depth evaluation yields the following insights. 1) Despite slower, patch-based models outperform pooling-based ones. 2) The D_L cue, i.e. CCNN configuration, allows for the best results when dealing with disparity maps generated by local approaches either conventional (CENSUS) or learning-based (MC-CNN). 3) For algorithms generating smoother disparity maps like SGM, the most effective configuration is PBCP coupled with the reference image. Nonetheless, CCNN is still competitive and the right disparity map required by PBCP is not always available, especially when dealing with end-to-end depth sensing networks. 4) In such a case, experiment on DispNetC maps highlight once again that CCNN is the best option among the considered ones, stressing the low contribution given by reference image, already exploited at its best to estimate the disparity map. 5) The same behaviour is also confirmed when tackling confidence estimation from monocular depth estimation models such as Monodepth.

In summary, processing the disparity map alone, as done by the original CCNN network [13], turns out the most versatile and overall effective strategy across all algorithms and datasets.

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

References

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

[2] M. Poggi, F. Tosi, and S. Mattoccia, "Quantitative evaluation of confidence measures in a machine learning world," in *The IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct 2017.

[3] J. L. Schönberger, S. Sinha, and M. Pollefeys, "Learning to Fuse Proposals from Multiple Scanline Optimizations in Semi-Global Matching," in *15th European Conference on Computer Vision (ECCV)*, 2018.

[4] M. G. Park and K. J. Yoon, "Leveraging stereo matching with learning-based confidence measures," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[5] A. Seki and M. Pollefeys, "Patch based confidence prediction for dense disparity map," in *Proceedings of the 27th British Conference on Machine Vision, BMVC*, 2016.

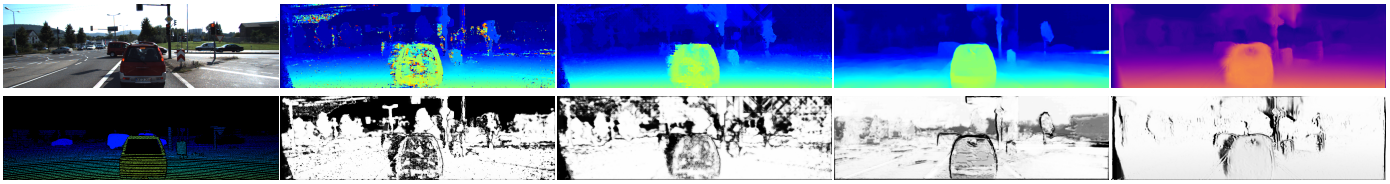

Fig. 3. **Qualitative results on KITTI 2015.** Leftmost side: reference frame (top) and ground truth disparity (bottom). Then, from left to right: (top) disparity maps for for MC-CNN-fst, SGM, DispNetC and inverse depth map for Monodepth, (bottom) corresponding best confidence estimation method for each one (respectively, CCNN, LF-PBCP, CCNN and CCNN). Disparity is encoded with colormap jet, inverse monocular depth map with colormap plasma, confidence with grayscale values.

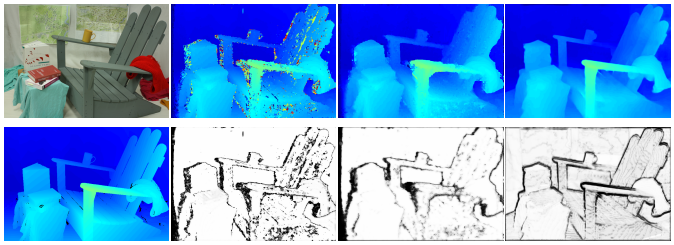

Fig. 4. **Qualitative results on Middlebury 2014.** Leftmost side: reference frame (top) and ground truth disparity (bottom). Then, from left to right, (top) disparity maps for MC-CNN-fst, SGM, DispNetC and (bottom) outcome of the best confidence estimation method for each one (CCNN). Disparity is encoded with colormap jet, confidence with grayscale values.

[6] M. Poggi and S. Mattoccia, "Learning a general-purpose confidence measure based on o(1) features and a smarter aggregation strategy for semi global matching," in *4th International Conference on 3D Vision (3DV)*, 2016.

[7] M. Poggi, F. Tosi, and S. Mattoccia, "Learning a confidence measure in the disparity domain from o(1) features," *Computer Vision and Image Understanding*, vol. 193, p. 102905, apr 2020.

[8] A. Tonioni, M. Poggi, S. Mattoccia, and L. Di Stefano, "Unsupervised adaptation for deep stereo," in *The IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct 2017.

[9] ——, "Unsupervised domain adaptation for depth prediction from images," *IEEE Transactions on Pattern Analysis and Machine Intelligence (accepted)*, 2019.

[10] F. Tosi, M. Poggi, A. Tonioni, L. Di Stefano, and S. Mattoccia, "Learning confidence measures in the wild," in *Proceedings of the 28th British Conference on Machine Vision, BMVC*, September 2017.

[11] G. Marin, P. Zanuttigh, and S. Mattoccia, "Reliable fusion of tof and stereo depth driven by confidence measures," in *14th European Conference on Computer Vision (ECCV)*, 2016, pp. 386–401.

[12] M. Poggi, G. Agresti, F. Tosi, P. Zanuttigh, and S. Mattoccia, "Confidence estimation for tof and stereo sensors and its application to depth data fusion," *IEEE Sensors Journal (accepted)*, 2019.

[13] M. Poggi and S. Mattoccia, "Learning from scratch a confidence measure," in *Proceedings of the 27th British Conference on Machine Vision, BMVC*, 2016.

[14] Z. Fu and M. Ardabilian, "Stereo matching confidence learning based on multi-modal convolution neural networks." in *Representation, analysis and recognition of shape and motion FroM Image data (RFMI)*, 2017.

[15] F. Tosi, M. Poggi, A. Benincasa, and S. Mattoccia, "Beyond local reasoning for stereo confidence estimation with deep learning," in *15th European Conference on Computer Vision (ECCV)*, September 2018.

[16] X. Chen, G. Y. Tian, J. Wu, C. Tang, and K. Li, "Feature-based registration for 3d eddy current pulsed thermography," *IEEE Sensors Journal*, vol. 19, pp. 6998–7004, 2019.

[17] M. Poggi and S. Mattoccia, "Learning to predict stereo reliability enforcing local consistency of confidence maps," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[18] S. Kim, S. Kim, D. Min, and K. Sohn, "Laf-net: Locally adaptive fusion networks for stereo confidence estimation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[19] R. Haeusler, R. Nair, and D. Kondermann, "Ensemble learning for confidence measures in stereo vision," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[20] A. Spyropoulos, N. Komodakis, and P. Mordohai, "Learning to detect ground control points for improving the accuracy of stereo matching." in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[21] M. Poggi, F. Tosi, and S. Mattoccia, "Even more confident predictions with deep machine-learning," in *12th IEEE Embedded Vision Workshop (EVW2017) held in conjunction with IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[22] S. Kim, D. Min, S. Kim, and K. Sohn, "Feature augmentation for learning confidence measure in stereo matching," *IEEE Transactions on Image Processing*, vol. 26, no. 12, Dec 2017.

[23] M. Poggi, F. Tosi, and S. Mattoccia, "Efficient confidence measures for embedded stereo," in *19th International Conference on Image Analysis and Processing (ICIAP)*, September 2017.

[24] C. Mostegel, M. Rumpler, F. Fraundorfer, and H. Bischof, "Using self-contradiction to learn confidence measures in stereo vision," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[25] H. Hirschmüller, "Accurate and efficient stereo processing by semi-global matching and mutual information," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[26] A. Spyropoulos and P. Mordohai, "Ensemble classifier for combining stereo matching algorithms," in *3rd International Conference on 3D Vision (3DV)*, Oct 2015.

[27] M. Poggi and S. Mattoccia, "Deep stereo fusion: combining multiple disparity hypotheses with deep-learning," in *4th International Conference on 3D Vision (3DV)*, 2016.

[28] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International journal of computer vision*, vol. 47, no. 1-3, pp. 7–42, 2002.

[29] H. Hirschmüller and D. Scharstein, "Evaluation of stereo matching costs on images with radiometric differences," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, pp. 1582–1599, 08 2008.

[30] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondence," in *Proceedings of the Third European Conference on Computer Vision (Vol. II)*, ser. 3rd European Conference on Computer Vision (ECCV). Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1994, pp. 151–158.

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

[32] M. Menze and A. Geiger, "Object scene flow for autonomous vehicles," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[33] D. Scharstein, H. Hirschmüller, Y. Kitajima, G. Krathwohl, N. Nesic, X. Wang, and P. Westling, "High-resolution stereo datasets with subpixel-accurate ground truth." in *GCPR*, ser. Lecture Notes in Computer Science, X. Jiang, J. Hornegger, and R. Koch, Eds., vol. 8753. Springer, 2014, pp. 31–42.

[34] J. Zbontar and Y. LeCun, "Computing the stereo matching cost with a convolutional neural network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1592–1599.

[35] K. Batsos, C. Cai, and P. Mordohai, "Cbmv: A coalesced bidirectional matching volume for disparity estimation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[36] F. Liu, C. Shen, G. Lin, and I. Reid, "Learning depth from single monocular images using deep convolutional neural fields," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 10, pp. 2024–2039, 2016.

[37] J. Zbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches," *Journal of Machine Learning Research*, vol. 17, no. 1-32, p. 2, 2016.

[38] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
[39] A. Kendall, H. Martirosyan, S. Dasgupta, P. Henry, R. Kennedy, A. Bachrach, and A. Bry, "End-to-end learning of geometry and context for deep stereo regression," in *The IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct 2017.
[40] Z. Liang, Y. Feng, Y. Guo, H. Liu, W. Chen, L. Qiao, L. Zhou, and J. Zhang, "Learning for disparity estimation through feature constancy," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
[41] C. Jia-Ren and C. Yong-Sheng, "Pyramid stereo matching network," *arXiv preprint arXiv:1803.08669*, 2018.
[42] F. Zhang, V. Prisacariu, R. Yang, and P. H. Torr, "Ga-net: Guided aggregation net for end-to-end stereo matching," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
[43] X. Guo, K. Yang, W. Yang, X. Wang, and H. Li, "Group-wise correlation stereo network," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
[44] A. Tonioni, F. Tosi, M. Poggi, S. Mattoccia, and L. Di Stefano, "Real-time self-adaptive deep stereo," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
[45] A. Tonioni, O. Rahnama, T. Joy, L. Di Stefano, A. Thalaiyasingam, and P. Torr, "Learning to adapt for stereo," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
[46] M. Poggi, D. Pallotti, F. Tosi, and S. Mattoccia, "Guided stereo matching," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
[47] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
[48] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *International Journal of Robotics Research (IJRR)*, 2013.
[49] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Advances in neural information processing systems*, 2014, pp. 2366–2374.

Matteo Poggi received Master degree in Computer Science and PhD degree in Computer Science and Engineering from University of Bologna in 2014 and 2018 respectively. Currently, he is a Post-doc researcher at Department of Computer Science and Engineering, University of Bologna. His research interests include deep learning for depth estimation and embedded computer vision.

Fabio Tosi received the Master degree in Computer Science and Engineering at Alma Mater Studiorum, University of Bologna in 2017. He is currently in the PhD program in Computer Science and Engineering of University of Bologna, where he conducts research in deep learning and depth sensing related topics.

Stefano Mattoccia received a Ph.D. degree in Computer Science Engineering from the University of Bologna in 2002. Currently he is an associate professor at the Department of Computer Science and Engineering of the University of Bologna. His research interest is mainly focused on computer vision, depth perception from images, deep learning and embedded computer vision.