Leveraging confident points for accurate depth refinement on embedded systems

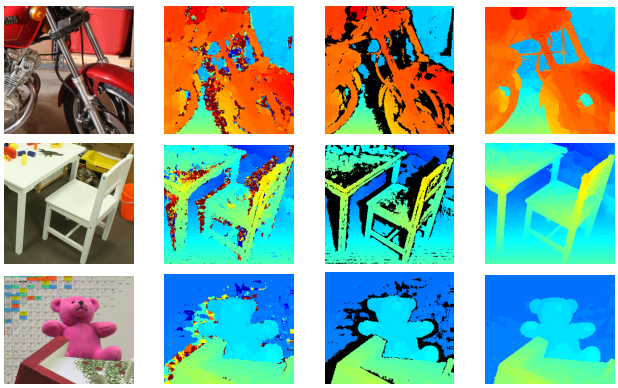
Fabio Tosi, Matteo Poggi, Stefano Mattoccia
Department of Computer Science and Engineering (DISI)
University of Bologna, Italy
{fabio.tosi5, m.poggi, stefano.mattoccia }@unibo.it

Abstract

Despite the notable progress in stereo disparity estimation, algorithms are still prone to errors in challenging conditions. Thus, heuristic disparity refinement techniques are usually deployed to improve accuracy. Moreover, state-of-the-art methods rely on complex CNNs requiring power hungry GPUs not suited for many practical applications constrained by limited computing resources. In this paper, we propose a novel technique for disparity refinement leveraging on confidence measures and a novel, automatic learning-based selection method to discard outliers. Then, a non-local strategy infers missing disparities by analyzing the closest reliable points. This framework is very fast and does not require any hand-tuned thresholding. We assess the performance of our Non-Local Anchoring (NLA), standalone refinement techniques and methods leveraging on confidence measures inside the stereo algorithm. Our evaluation with two popular stereo algorithms shows that our proposal significantly ameliorates their accuracy on Middlebury v3 and KITTI 2015 datasets. Moreover, since our method relies only on cues computed in the disparity domain, it is suited even for COTS stereo cameras coupled with embedded systems, e.g. *nVidia Jetson TX2.*

1. Introduction

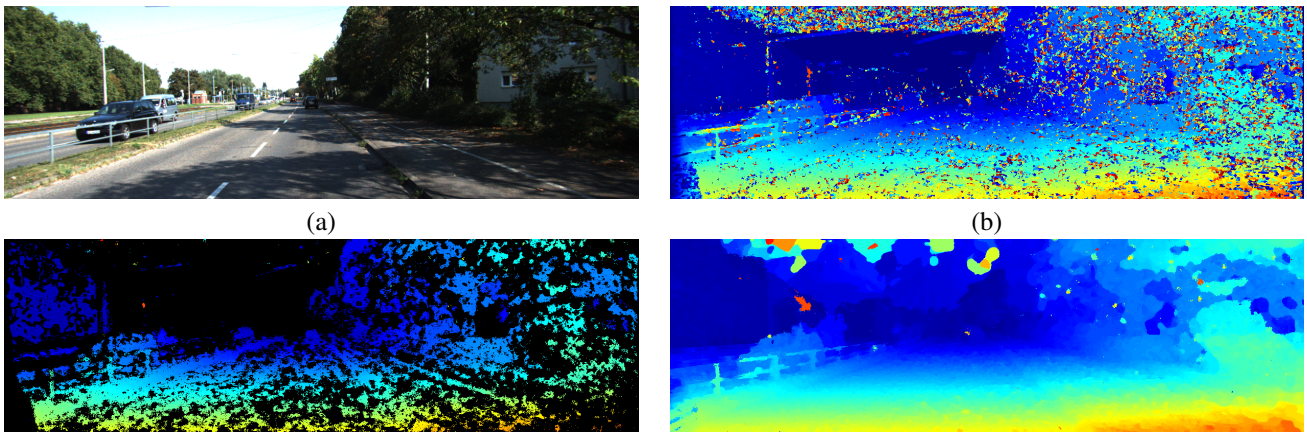
Stereo is one of the most popular technique to infer depth from two or more images and challenging datasets, such as KITTI [5, 17] and Middlebury [28], clearly emphasized that it is still an open problem. State-of-the-art algorithms [3, 12] require expensive and power-hungry GPUs to run in a reasonable amount of time, making them unsuited for many practical applications constrained by hardware resources or energy consumption. Conventional (*i.e.* pre-deep learning) algorithms still achieve accurate results leveraging on multi-step pipelines, each one contributing to increasing the overall effectiveness with different degrees of reliability. A notable example is the Semi-Global Matching algorithm (SGM) [7], implemented in many variants thanks to its trade-off between accuracy and complexity, that usually deploy interpolation and refinement steps on estimated disparity maps. In their seminal work, Zbontar and LeCun [43] showed how plugging deep learning into a conventional stereo SGM pipeline yielded very accurate results on KITTI and Middlebury datasets not far from end-to-end networks [3, 12].



(a) (b) (c) (d)

Figure 1. Non-Local Anchoring framework applied to three Middlebury v3 stereo pairs. From top to bottom: *MotorcycleE*, *PianoL*, *Teddy*. (a) Detail of left image, (b) raw disparity map, (c) set of reliable pixels according to an ideal confidence measure, (d) refined disparity map.

One of the steps involved, referred to as *disparity refinement*, attempts to recover errors from the disparity map. While some refinement procedures rely on simple filters (*e.g.*, median or bilateral filters) others exploit cues from the disparity map and the input stereo pair. Confidence measures allow to detect unreliable matches produced by stereo algorithms and, recently, strategies based on machine-learning achieved state-of-the-art results [26]. Confidence measures have been deployed in different steps of stereo pipelines, with the aim to further improve the overall accuracy. In this paper, we propose *Non-Local Anchoring* (NLA), a novel disparity refinement method relying on confidence measures, outlined in Figure 1. Given a disparity map generated by any stereo algorithm, the confidence



(a) (b) (c) (d)

Figure 2. NLA in action on KITTI 2015 dataset [17]. (a) Left frame from stereo pair 000027, (b) raw disparity map computed by the Block-Matching algorithm (BM) algorithm, (c) sparse disparity map containing assumed reliable points RP, (d) refined disparity map with our proposal.

measure allows us to detect and removing erroneous pixels. Then, for each discarded pixel, among the remaining *reliable points* (RP) a subset of *anchors*, not necessarily in a close neighborhood of the examined pixel, is chosen to infer, according to both spatial and color information from the reference image, a new disparity value. Moreover, a CPU-friendly machine-learning framework based on a random forest classifier is proposed to deal with automatic identification of unreliable disparity assignments by analyzing local and global properties of the confidence on the whole image. This novel strategy allows us to remove the need for a heuristic selection of a confidence threshold often carried-out in this field [31, 29].

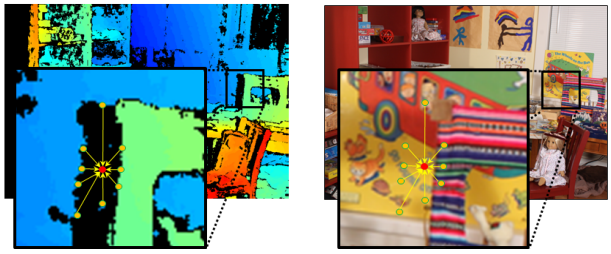
To assess the effectiveness of our proposal, we report an extensive evaluation on the Middlebury v3 dataset comparing our framework to conventional disparity refinement methodologies as well as with recently proposed confidence-based approaches, acting on the Disparity Space Image (DSI) [27] also referred to as the *cost-volume*. Differently, NLA acts in the disparity domain hence does not require at all the cost volume that might be not available in some circumstances, *e.g.* when dealing with a commercial off-the-shelf (COTS) stereo camera. Factors like the number of anchors deployed and a further local aggregation strategy included in our proposal will be discussed and compared. Moreover, we evaluate our framework also on KITTI 2015 dataset [17] to further confirm the effectiveness of our method on indoor and outdoor data. Figure 2 shows the outcome of our proposal on the frame 166 of the KITTI 2015 dataset deploying the disparity map generated by the popular Block-Matching (BM) algorithm.

Finally, we point out how the proposed strategy works by acting in the disparity domain only with reduced computational complexity, fitting very well with COTS stereo cameras and in general with embedded devices, such as nVidia Jetson TX2 used to measure runtime in our experiments.

2. Related work

Confidence measures The confidence measure literature is relevant to our work. Initially Hu and Mordohai [8] carried out an extensive taxonomy and evaluation of confidence measures, categorizing them according to the processed cues. They also proposed a common metric to compare the effectiveness of different confidence measures and assess their performance to detect uncertain disparity assignments. More recently, machine learning techniques have been used to infer more effective confidence measures. Hausler et al. [6] deployed, for the first time, a random forest combining different (as much as possible) orthogonal measures and features aimed at classifying each pixel as a correct or wrong match. Improved methods based deploying a random forest were proposed in [31, 20, 23]. Recent works deployed CNNs to infer confidence as well. In particular in [24] by only processing the left disparity map, in [29] using as input cue the left and right maps and in [36] exploiting local and global cues. Finally, confidence measures have been effectively deployed as cues to improve the overall accuracy of stereo within conventional pipelines as recently shown in [29, 23, 20, 31]. Poggi et al. [26] evaluated conventional and learning-based confidence measures highlighting how the learned-ones are much more effective. Finally, we point out that unsupervised learning of confidence measures has been tackled in [18] and [37].

Pre-deep learning stereo. The extensive literature concerning stereo has been enriched, in the years, by several contributions and novel methodologies. Despite that, the majority of them (with rare exceptions such as [16]) can be still categorized according to the taxonomy proposed



(a) (b)

Figure 3. Overview of NLA on *Playtable* image from Middlebury v3. (a) Disparity map containing reliable RP points only, (b) reference image. For each unreliable pixel (red), anchors (yellow) are selected as the closest RPs along different directions.

by Scharstein and Szeliski [27], which also lists common steps: *cost computation*, *cost aggregation*, *disparity computation* and *disparity refinement*. Algorithms focusing on the first two steps are referred to as *local* while *global* methods mainly rely on the optimizing steps. The latter ones are usually more effectively but computationally more expensive. Semi-Global Matching (SGM) proposed by Hirschmuller [7] represents a good trade-off between performance and accuracy making it one of the most deployed solutions to infer dense disparity maps in practical applications. Disparity refinement techniques are usually deployed on top of traditional stereo pipelines to push accuracy toward optimality further. Common approaches consist of image filtering operations, like median filter [21], weighted median filter [44], guided filter [40], bilateral filter [33] or the recent fast bilateral solver [2]. One of the most effective and used refinement technique consists in a *left-right consistency* check to detect both occlusions and mismatches, *interpolating* the former points with the background and the latter ones with a median from nearby disparity. Such strategy [7, 43] is often coupled with median and bilateral filters.

Deep learning stereo. The recent spread of deep learning was applied to stereo as well, with [42, 43] being the first one to tackle matching cost computation employing a Convolutional Neural Network (CNN) working on image patches. Other authors followed this path addressing efficiency [4, 13]. In [22] a CNN is deployed to combine multiple out-of-the-box stereo matchers to obtain more accurate results, inspired by the work of Spyropolous and Mordohai [32] which carried it out by using a random forest. Conversely, Mayer et al. [16] proposed DispNet, the first deep architecture for end-to-end stereo computation, completely departing from conventional stereo methodologies. Although this method is not the most accurate one on KITTI and Middlebury datasets, it represents a ground-breaking approach to tackle stereo. Since then, the study of end-to-end architectures for dense stereo matching became dominant. Kendall et al. [10] introduced 3D CNNs to regularize a volume obtained through concatenation between left and right features on the disparity axis. Further works improve the results of this approach [3, 39, 11] or 2D networks [19, 12], optionally learning stereo jointly with other tasks [38, 30, 9]. Some works addressed domain shift issues affecting the aforementioned frameworks, by either adapting to new environments offline [34] or online [35], as well as leveraging the guide sourced by external sensors [25].

Although CNNs represent the preferred choice in terms of accuracy for both stereo and confidence estimation, often their hardware requirements overwhelm the resources available in many embedded systems.

3. Non-Local Anchoring

In this section, we introduce the proposed NLA framework, that given a disparity map $\mathcal{D}$ and a confidence map $\mathcal{C}$ encoding the uncertainty of each pixel (the higher the confidence, the better the assumed reliability), infers a completely dense and more accurate map. It starts by classifying each disparity point belonging to $\mathcal{D}$ in two categories: *reliable* and *unreliable* points, for short RP and UP respectively. In literature [31, 29], this task is accomplished by setting a threshold value ξ and considering as RP the points with a confidence value higher than ξ. That is,

$$RP = \{p \in \mathcal{D}, \mathcal{C}(p) \geq \xi\} \tag{1}$$

consequently, the remaining ones are considered UP,

$$UP = \{p \in \mathcal{D}, \mathcal{C}(p) < \xi\} \tag{2}$$

A new disparity map $\mathcal{D}'$ is then obtained by removing from $\mathcal{D}$ the UP set. The resulting $\mathcal{D}'$ map is characterized by a lower error rate, ideally 0, at the cost of a sparser distribution of pixels compared to $\mathcal{D}$.

Afterward, the full density of $\mathcal{D}'$ is restored by looking at reliable information within the RP set. To do so, given a pixel p and a 2D vector d, we first define a subset of pixels $P(p, d)$ as the *path* on which p lays according to the direction of d:

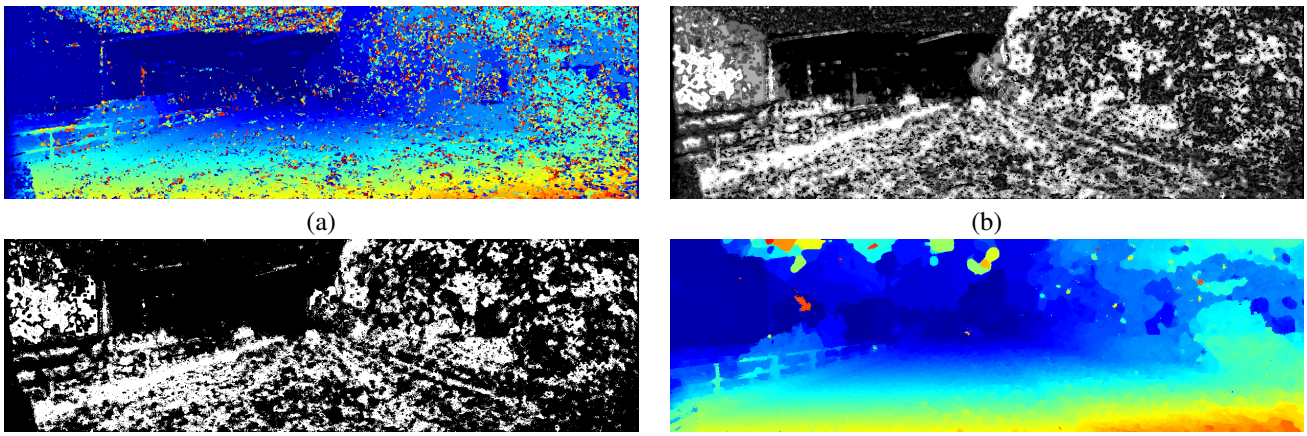
$$P(p, d) = \{q \in \mathcal{D}, \alpha \in N, q = p + \alpha d\} \tag{3}$$

For a pixel $u \in$ UP, we define its *anchor* along direction d, as the closest pixel to u laying on path $P(u, d)$:

$$a(u, d) = \{v \in RP, \min_{v} |u - v|\} \tag{4}$$

Given a set of paths on which u lays, a set $\mathcal{A}(u)$ of anchors will contribute to computing the new disparity value for such pixel. In particular, each anchor $a \in \mathcal{A}(u)$ spreads its disparity to u, weighting it according to a similarity function between features $\mathcal{I}(u)$ and $\mathcal{I}(a)$ as follows:

$$w(u, a) = \mathcal{G}(|\mathcal{I}(u) - \mathcal{I}(a)|) \cdot \mathcal{G}(|u - a|) \tag{5}$$



(a) (b) (c) (d)

Figure 4. RP selection. (a) Noisy disparity map computed by Block-Matching algorithm (BM) on stereo pair 000027 of the KITTI dataset, (b) O1 [23] confidence map, (c) set of RP (white) and UP (black) according to O1, (d) refined disparity map with our proposal.

The cues collected by each anchor $\mathcal{A}(u)$ are used to build a weighted histogram, on which each $w(u, a)$ increases the index corresponding to disparity hypothesis of pixel a. Finally, the weighted median is computed among the collected contributions:

$$\mathcal{D}(u) = \min_{k} \sum_{i=0}^{k} w(u, a_i) \geq \frac{1}{2} \sum_{i=0}^{n} w(u, a_i) \qquad (6)$$

We rely on a Gaussian function $\mathcal{G}$ to encode the similarity between the unreliable pixel and one of its anchor points and on color intensity $\mathcal{I}(u)$ in the reference image. This strategy, coupled with the weighted median, enables edge-preserving disparity propagation. Figure 3 shows an example of anchoring for an unreliable pixel (red), receiving the contribution from a set of anchors (yellows).

Computational complexity for NLA is extremely low, as all the anchors of each unreliable pixel and their corresponding weights can be processed on a single image scan for each path in constant time. It only depends on the size of the image and the number of paths deployed for anchoring. It is worth observing that our proposal, conversely from other methods, is not constrained to a restricted area (*i.e.*, local patches). Moreover, differently from recent methodologies exploiting confidence to improve stereo accuracy [31, 20, 23, 29], our framework acts on the disparity domain hence not requiring any information from the DSI thus enabling, for instance, its deployment with COTS devices.

Optionally, before replacing the unreliable pixel u according to the outlined strategy, a further local aggregation step can improve the effectiveness of the information gathered from nearby points. This optional phase can be carried out by building a DSI with the $w(u, a_k)$ weights and filtering it according to the same similarity function $\mathcal{G}$. This step enables the collection of additional contributions from nearby UP pixels q_k, which set of anchors $\mathcal{A}(q_k)$ is different from $\mathcal{A}(u)$.

4. Threshold-free RP selection

According to the description reported in Section 3, classifying the disparity values in UP and RP plays a key-role for NLA to achieve optimal performance. Thus, choosing the confidence threshold ξ is of paramount importance. This strategy is common to other successful attempts to exploit confidence measures inside stereo algorithms [31, 29] or, in general, when we want to remove erroneous matches from the disparity map. For such tasks, proper tuning of the threshold ξ is required to achieve the best results.

To address this issue, we propose a second level framework to effectively distinguish pixels into RP and UP according to features extracted from the confidence map without any manual tuning. To this aim, we fed to a random forest, trained in classification mode, the following local and global features computed from the confidence map:

- $\mathcal{C}_p$, the confidence value for pixel p
- $\mu_N(\mathcal{C}_p)$, the average confidence computed on a local window $\mathcal{N}$, centered in p and made of $\bar{\bar{N}}$ pixels

$$\mu^{\mathcal{C}}_{\mathcal{N}}(p) = \frac{1}{\bar{\bar{N}}} \sum_{q \in \mathcal{N}} \mathcal{C}_q \qquad (7)$$

- $\sigma_N(\mathcal{C}_p)$, the variance of confidence on a local window $\mathcal{N}$, centered in p and made of $\bar{\bar{N}}$ pixels

$$\sigma^{\mathcal{C}}_{\mathcal{N}}(p) = \frac{1}{\bar{\bar{N}}} \sum_{q \in \mathcal{N}} [\mathcal{C}_q - \mu_{\mathcal{N}}(\mathcal{C}_p)]^2 \qquad (8)$$

Stereo algorithm	All				Non-occ			
	bad 1(%)	bad 2(%)	RMSE	MAE	bad 1(%)	bad 2(%)	RMSE	MAE
BM	35.13	32.32	14.32	6.85	26.37	23.54	10.94	4.49
+ FBS [2]	33.47	28.60	12.79	5.28	24.67	19.59	8.32	2.93
+ MF [21]	27.43	23.99	10.14	4.32	18.42	14.95	5.82	2.17
+ WMF [44]	26.22	22.92	10.08	4.18	17.22	13.91	5.56	2.00
+ WMF + GF [40]	26.33	22.92	11.27	4.75	17.41	14.04	7.59	2.86
+ WMF + JBF [40]	28.03	24.93	10.95	4.68	18.86	15.75	6.87	2.44
+ LRI [43]	27.99	24.99	19.10	6.97	20.64	17.81	14.93	4.51
+ LRI + MF + BF [43]	26.02	21.53	15.78	5.94	18.68	14.23	11.18	3.58
+ LC [14]	24.23	20.00	11.68	4.74	16.30	12.23	7.93	2.65
+ NLA + O1	**22.90**	**19.86**	**9.36**	**3.63**	**14.08**	**11.20**	**5.33**	**1.71**
+ *NLA + opt.*	*6.23*	*4.07*	*3.06*	*0.85*	*2.20*	*1.21*	*1.77*	*0.44*

Table 1. Experimental results averaged on Middlebury v3 with BM algorithm. Best results are in bold.

- $\delta_\mu(p)$, or *deviation from average confidence*, the absolute difference between $\mathcal{C}(p)$ and the average confidence over the entire disparity map $\mathcal{D}$ (*i.e.*, $\mu_\mathcal{D}(\mathcal{C})$)

$$\delta_\mu(p) = |\mathcal{C}_p - \mu^\mathcal{C}_\mathcal{D}(\mathcal{C})| \tag{9}$$

- $\delta_\sigma(p)$, or *deviation from variance of confidence*, the absolute difference between $\mathcal{C}(p)$ and the average confidence over the entire disparity map $\mathcal{D}$ (*i.e.*, $\sigma_\mathcal{D}(\mathcal{C})$)

$$\delta_\sigma(p) = |\mathcal{C}_p - \sigma^\mathcal{C}_\mathcal{D}(\mathcal{C})| \tag{10}$$

Concerning μ and σ, we process these features three times with increasing size of the local window $\mathcal{N}$, respectively $\Omega = 3 \times 3$, $\Theta = 7 \times 7$ and $\Gamma = 11 \times 11$. As result, we obtain the following feature vector $f_9(p)$

$$f_9(p) = \{\mathcal{C}_p, \mu^\mathcal{C}_\Omega(p), \mu^\mathcal{C}_\Theta(p), \mu^\mathcal{C}_\Gamma(p), \sigma^\mathcal{C}_\Omega(p), \sigma^\mathcal{C}_\Theta(p), \sigma^\mathcal{C}_\Gamma(p), \delta_\mu(p), \delta_\sigma(p)\} \tag{11}$$

We train on such feature vector a random forest, made of 10 trees, maximum depth equal to 15 and a minimum number of samples in each node to split equal to 12, in order to achieve an automatic RP selection without any hand-chosen threshold. Figure 4 shows a qualitative example of RP selection. Given a disparity map (a) and a confidence map (b), the reliable pixels are selected (c) and plugged into the NLA framework to obtain the final map (d).

5. Experimental results

In this section, we evaluate the effectiveness of the proposed NLA framework with disparity maps obtained, on challenging datasets, by two stereo algorithms:

- Block Matching (BM), a local method computing matching costs on a 5×5 census transformed image [41], locally aggregated by a 5×5 box filter
- SGM [7], using as data term the normalized aggregated costs as BM algorithm and penalty parameters P1 and P2 tuned to 0.2 and 0.5

The choice was driven by the fast inference enabled by the two algorithms. Embedded stereo cameras with onboard processing (*e.g.*, [1] or [15]) can run both BM and SGM at more than 30 FPS, sourcing disparity estimates in real-time with limited power consumption. In such a scenario, NLA can further improve the overall accuracy with low complexity, making it suited for embedded systems.

To exhaustively assess the effectiveness of our proposal, we compare it to state-of-the-art disparity refinement methods acting in the disparity domain. Moreover, since NLA relies on a confidence measure, we also compare it with recent methodologies exploiting confidence prediction to improve stereo accuracy [20, 23, 29] acting in the DSI domain. We also evaluate for NLA the effect yielded by a different number of anchors and by the optional aggregation step outlined. Moreover, we validate the effectiveness of UP/RP selection module by reporting comparison with the manual optimal choice of the ξ value by cross-validation. We evaluate all these aspects on the Middlebury v3 [28] training dataset and then we evaluate the effectiveness of the overall NLA framework also on KITTI 2015 [17].

5.1. Evaluation on Middlebury v3

In this section, we provide exhaustive experimental results concerning the full NLA framework (*i.e.*, deploying the random forest for threshold-free RP selection and local aggregation step) and other refinement methods on the Middlebury v3 training dataset[1].

Comparison with other refinement strategies. In tables 1 and 2 we report results achieved by the following disparity refinement methods: fast bilateral solver (FBS [2]), median filter (MF [21]), weighted median filter (WMF

[1]We process Middlebury v3 stereo pairs at quarter resolution. All the results reported in this paper have been computed at such resolution on training split.

Stereo algorithm	All				Non-occ			
	bad 1%	bad 2%	RMSE	MAE	bad 1%	bad 2%	RMSE	MAE
SGM [7]	24.38	22.00	13.18	5.33	14.52	12.14	7.96	2.49
+ FBS [2]	25.06	21.55	12.05	4.51	15.46	11.93	6.80	2.10
+ MF [21]	23.13	20.44	11.20	4.43	13.45	10.74	5.95	1.91
+ WMF [44]	21.88	19.29	11.32	4.34	12.26	9.67	5.69	1.74
+ WMF + GF [40]	22.22	19.56	12.54	4.96	12.72	10.10	7.96	2.68
+ WMF + JBF [40]	22.25	19.80	11.94	4.61	12.46	10.02	6.45	1.91
+ LRI [43]	21.46	18.77	14.12	4.84	13.24	10.74	8.63	2.34
+ LRI + MF + BF [43]	22.01	17.68	13.26	4.81	14.09	9.81	7.80	2.40
+ LC [14]	20.39	16.60	10.56	3.98	12.59	9.05	6.56	1.95
+ Lev.stereo* [20]	22.22	19.52	12.45	4.60	13.38	10.73	7.39	2.20
+ Lev.stereo [20]	21.69	18.66	13.63	3.74	13.63	10.05	6.13	1.96
+ Smart-SGM [23]	22.67	19.71	11.51	4.33	13.57	10.78	6.54	2.05
+ PBCP* [29]	23.97	21.56	19.36	6.54	14.12	11.72	12.07	3.10
+ PBCP [29]	23.72	21.31	18.79	6.34	13.89	11.49	11.47	2.97
+ NLA + O1	**18.68**	**15.44**	**7.16**	**2.65**	**11.94**	**9.29**	**4.59**	**1.45**
+ NLA + opt.	*7.72*	*5.18*	*3.50*	*0.99*	*3.24*	*1.81*	*2.01*	*0.49*

Table 2. Experimental results averaged on Middlebury v3 with SGM [7] algorithm. Best results are in bold. Algorithms marked with * use the original confidence measure proposed in the paper.

Stereo algorithm	All				Non-occ			
	bad 1(%)	bad 2(%)	RMSE	MAE	bad 1(%)	bad 2(%)	RMSE	MAE
SGM [7]	24.38	22.00	13.18	5.33	14.52	12.14	7.96	2.49
+ NLA + O1 ($\xi = 0.4$)	18.90	15.86	8.06	2.99	**11.44**	**8.77**	4.63	**1.44**
+ NLA + O1 (ξ-less)	**18.68**	**15.44**	**7.16**	**2.65**	11.94	9.29	**4.59**	1.45

Table 3. Experimental results on Middlebury v3 with SGM, comparing results obtained by NLA when using a threshold or the random forest classification of the RP. Best results in bold.

[44]), weighted median filter together plus guided filter (WMF + GF [40]), weighted median filter plus joint bilateral filter (WMF + JBF [40]) and local consistency filter (LC [14]). All of these methods process only disparity map and the reference image. For each of these methods the patch size is set to 15×15. Moreover, we include left right interpolation (LRI) and the full refinement pipeline deployed in [43] (LRI + MF + BF) using authors'. We report, for each method, the amount of pixels having a disparity error larger than 1 and 2 (bad 1% and bad2%), as well as root mean square error (RMSE) and mean average error (MAE). In the same tables, we show results concerning the NLA framework with 16 anchors (*i.e.*, from horizontal, vertical, diagonal and half-diagonal directions) using the state-of-the-art O1 [23] confidence measure. It is obtained by training a random forest framework to process 20 features extracted from the disparity map, that are Disparity Agreement (DA), Disparity Scattering (DS), Median Deviation of Disparity (MDD), Median disparity (MED) and Variance of disparity (VAR) on four windows of size $5 \times 5, 7 \times 7, 9 \times 9$ and 11×11 [23]. Its effectiveness drove the choice of this measure in the estimation of correct matches and by the aim of our framework, working in the disparity domain only and possibly running on constrained architectures, for which deep learning approaches [24, 29, 36] would not be suited. We followed implementation notes, hyper-parameters tuning and code provided by the authors [23], training on a subset of images from KITTI 2012 dataset (the first 20 images) as in [26]. Since the effectiveness of the confidence measure is crucial for our method, we also report in the final row the results achieved by NLA processing an optimal confidence measure, capable of ideally distinguish between RP and UP. This represents the lower bound for the error rate with NLA. The automatic selection method proposed was trained on the 13 additional images available in Middlebury v3 dataset [28] for each of the two considered algorithms. Table 1 report the effectiveness of disparity refinement methods with the BM algorithm. We can notice how the proposed NLA outperforms all of the considered refinement methods. In particular, compared to the second best method LC, NLA is more effective by nearly 2% on both all pixels and non-occluded. The last row highlights how, if an ideal confidence measure is deployed, our framework is capable of reducing the error rate from over 35% of wrong pixels in the image to almost 6%. Table 2 shows the results with SGM [7]. Since our SGM implementation is based on BM algorithm to obtain the data term, we first highlight how the results obtained by processing maps by NLA are very similar (even better in this case) to those obtained by running SGM optimization on the entire DSI (without applying any

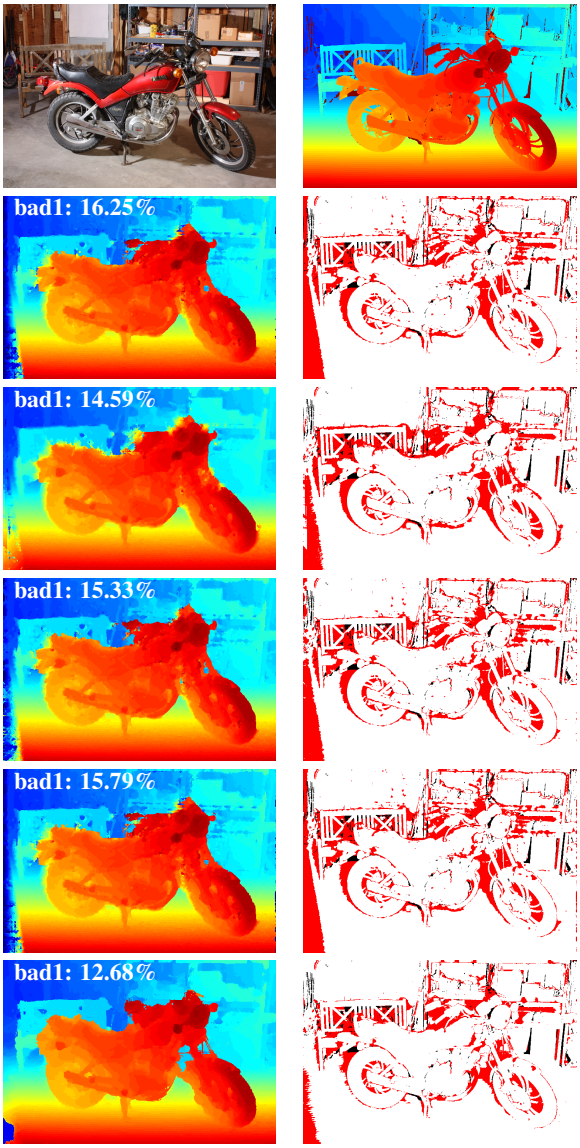


Figure 5. Qualitative results on Motorcycle stereo pair. First row: reference image and ground-truth disparity. Then, from top to bottom, disparity maps with overimposed bad1 rate and error maps for, respectively, SGM [7], SGM+Lev.stereo [20], Smart-SGM [23], SGM+PBCP [29] and SGM+NLA. All methods use O1 as confidence measure.

additional post-processing step, not deployed on our baseline SGM). This proves the effectiveness of our proposal when compared to more complex approaches such as SGM. Moreover, the DSI of the filtering map is not required with NLA, while SGM necessarily needs such information. In these experiments, we also deploy three additional methodologies relying on confidence measures to improve the results of SGM. The first one is a confidence-based modulation of the DSI carried-out before the SGM optimization, referred to as *Lev.stereo* [20]. The second one is a weighted

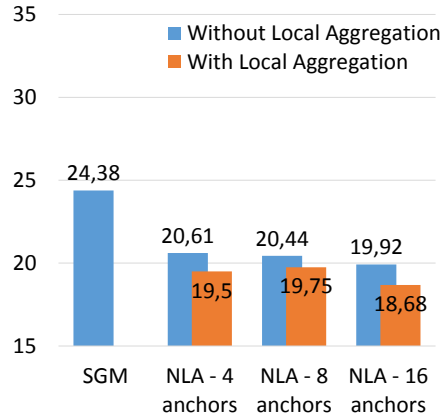


Figure 6. Experimental results on the entire Middlebury v3 dataset, varying the number of anchors and enabling/disabling local aggregation with NLA framework, SGM algorithm + O1.

sum of the contribution of the different scanlines, according to confidence, referred to as *Smart-SGM* [23]. The last one consists of a dynamic setting of the smoothness terms P1 and P2 according to confidence, referred to as *PBCP* [29]. We included them as representative state-of-the-art methodologies relying on confidence measures to improve the accuracy of stereo and we report results obtained when processing the confidence measures they were proposed with (marked with * in the table) as well as with the same one deployed by NLA for a fair comparison. We can observe how the NLA framework outperforms all of them, obtaining its best accuracy deploying the O1 measure. Moreover, our proposal works in the disparity domain, not requiring intermediate results from the SGM pipeline and it is thus a general-purpose technique suited for any stereo algorithm. Figure 5 shows a qualitative comparison between considered approaches and NLA.

Evaluation of RP selection. Once confirmed the superiority of the full NLA framework, in this section we inquire about the effectiveness of the threshold-free RP selection enabled by the random forest classifier. Table 3 shows comparison between the results achieved by the manually selected threshold through k-fold cross-validation, highlighting how the random forest selection strategy increases, on average, the accuracy of the refined disparity maps when considering all pixels, while it performs slightly worse on non-occluded pixels, thus mainly improving selection and refinement occluded regions.

Ablation studies on NLA and runtime. To better understand the key factors enabling for such improvements, we report results concerning the use of a different amount of anchors as well as without the optional local aggregation step, deployed during the previous evaluations. Figure 6 plots the error rate as a function of the number of anchors

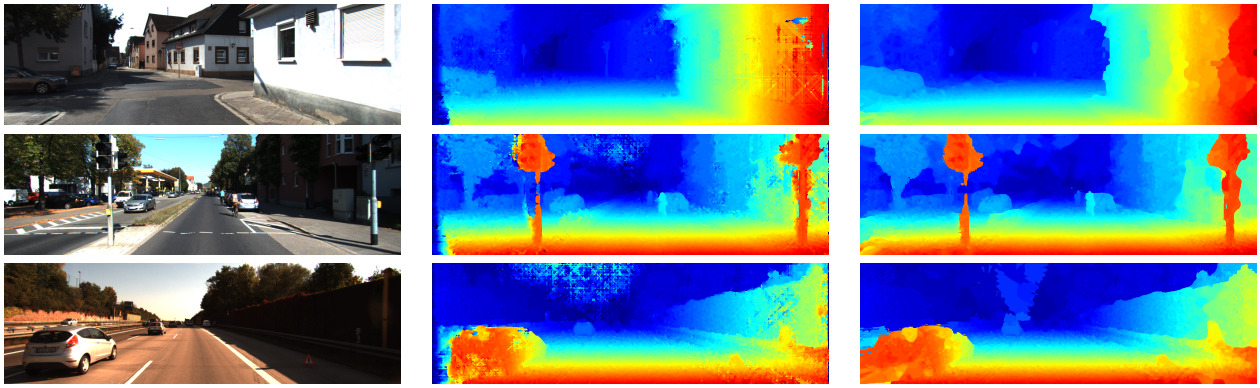


Figure 7. Qualitative results on KITTI 2015 dataset [17]. From top to bottom, stereo pairs 085, 186 and 197. From left to right, reference frame and disparity maps from SGM [7] or refined with NLA.

Stereo algorithm	bad 3% - All	
	BM	SGM
Baseline	37.30	10.78
MF [21]	19.95	8.73
WMF [44]	21.03	8.81
LRI [43]	25.29	10.12
LRI +MF + BF [43]	18.90	9.11
LC [14]	14.92	9.72
+ Lev.stereo* [20]	-	10.10
+ Lev.stereo [20]	-	9.52
+ Smart-SGM [23]	-	8.47
+ PBCP* [29]	-	10.63
+ PBCP [29]	-	10.62
NLA + O1	**11.42**	**7.68**

Table 4. Experimental results averaged on KITTI 2015 with BM and SGM [7] algorithms. Best results are in bold.

(4, 8 and 16) of the vanilla NLA framework (blue) and NLA with local aggregation (orange). It shows how the aggregation step enables a notable improvement, reducing the error rate by about 1% on SGM. About runtime, on a Jetson TX2 CPU (Arm v8), NLA runs in 1.82s without aggregation, rising up to 6.39s with full (not optimized) aggregation.

5.2. Evaluation on KITTI 2015

In this section, we report experimental results concerned with the KITTI 2015 training dataset [17], depicting outdoor environments very different from the Middlebury indoor scenes. We deploy for these experiments our full pipeline with 16 anchors, local aggregation and threshold-free selection of RP. Table 4 reports experimental results when refining disparity maps obtained by BM and SGM algorithms. We report the amount of pixels having a disparity error larger than 3 (bad 3%). Since KITTI 2015 dataset is very different compared to Middlebury v3, we tuned P1 and P2 smoothing penalties to 0.3 and 3 in order to obtain the most accurate results from the original SGM algorithm. We compare our results with best methods MF, WMF and LC approaches. We can observe how, even on this very different dataset, the NLA framework can reduce the error rate of the raw disparity maps by nearly 26% (BM) and by more than 3% (SGM), notably outperforming the other refinement techniques. Since the scene contents depicted by KITTI 2015 are more smooth compared to indoor scenes considered before (*e.g.*, large road planes), the smoothing constraint enforced by SGM is stronger than the non-local refinement processed by NLA, being nonetheless capable of reaching with BM a comparable degree of accuracy with significantly lower computational efforts. Focusing entirely on SGM results, we report, as for the Middlebury v3 evaluation, the improvements yielded by state-of-the-art confidence-based cost modulations proposed in [20, 23, 29]. Similarly to Middlebury v3, we evaluated the three previous strategies with their originally proposed confidence measures as well as with the same plugged into NLA for a fair comparison. The trend previously highlighted is confirmed on KITTI 2015 as well. Figure 7 shows additional qualitative results on KITTI 2015, comparing raw disparity maps by BM and SGM with those refined through NLA.

6. Conclusions

In this paper, we proposed a fast, yet accurate, non-local disparity refinement technique based on confidence measures. It jointly enables the benefits of techniques acting in the disparity domain and the power of confidence measures extracted from the same domain. Conversely from other similar techniques, leveraging on confidence measures and designed for specific algorithms, our proposal acts outside the stereo pipeline, making it a general purpose alternative, hence totally agnostic to the stereo algorithm generating disparity maps. Experimental results on popular datasets confirmed the superiority of NLA compared to known techniques when dealing with disparity maps obtained from algorithms suited for deployment on embedded devices.

References

[1] Intel Real Sense camera. https://realsense.intel.com/. 5

[2] T. Barron and B. Poole. The fast bilateral solver. In *Proceedings of the 14th European Conference on Computer Vision*, ECCV, 2016. 3, 5, 6

[3] J.-R. Chang and Y.-S. Chen. Pyramid stereo matching network. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1, 3

[4] Z. Chen, X. Sun, L. Wang, Y. Yu, and C. Huang. A deep visual correspondence embedding model for stereo matching costs. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 972–980, 2015. 3

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

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

[7] H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 30(2):328–341, feb 2008. 1, 3, 5, 6, 7, 8

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

[9] E. Ilg, T. Saikia, M. Keuper, and T. Brox. 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. 3

[10] 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 International Conference on Computer Vision (ICCV)*, Oct 2017. 3

[11] S. Khamis, S. Fanello, C. Rhemann, A. Kowdle, J. Valentin, and S. Izadi. Stereonet: Guided hierarchical refinement for real-time edge-aware depth prediction. In *15th European Conference on Computer Vision (ECCV 2018)*, 2018. 3

[12] 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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1, 3

[13] W. Luo, A. G. Schwing, and R. Urtasun. Efficient Deep Learning for Stereo Matching. In *Proc. CVPR*, 2016. 3

[14] S. Mattoccia. A locally global approach to stereo correspondence. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on. IEEE*, ICCV, 2009. 5, 6, 8

[15] S. Mattoccia and M. Poggi. A passive rgbd sensor for accurate and real-time depth sensing self-contained into an fpga. In *Proceedings of the 9th International Conference on Distributed Smart Cameras*, pages 146–151. ACM, 2015. 5

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

[17] M. Menze and A. Geiger. Object scene flow for autonomous vehicles. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 1, 2, 5, 8

[18] C. Mostegel, M. Rumpler, F. Fraundorfer, and H. Bischof. Using self-contradiction to learn confidence measures in stereo vision. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[19] J. Pang, W. Sun, J. S. Ren, C. Yang, and Q. Yan. Cascade residual learning: A two-stage convolutional neural network for stereo matching. In *The IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2017. 3

[20] M.-G. Park and K.-J. Yoon. Leveraging stereo matching with learning-based confidence measures. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 2, 4, 5, 6, 7, 8

[21] S. Perreault and P. Hbert. Median filtering in constant time. *IEEE Transactions on Image Processing*, 16(9):2389–2394, 2007. 3, 5, 6, 8

[22] M. Poggi and S. Mattoccia. Deep stereo fusion: combining multiple disparity hypotheses with deep-learning. In *Proceedings of the 4th International Conference on 3D Vision, 3DV*, 2016. 3

[23] 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 *Proceedings of the 4th International Conference on 3D Vision, 3DV*, 2016. 2, 4, 5, 6, 7, 8

[24] M. Poggi and S. Mattoccia. Learning from scratch a confidence measure. In *Proceedings of the 27th British Conference on Machine Vision, BMVC*, 2016. 2, 6

[25] M. Poggi, D. Pallotti, F. Tosi, and S. Mattoccia. Guided stereo matching. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3

[26] M. Poggi, F. Tosi, and S. Mattoccia. Quantitative evaluation of confidence measures in a machine learning world. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 1, 2, 6

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

[28] D. Scharstein and R. Szeliski. High-accuracy stereo depth maps using structured light. In *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, CVPR'03, pages 195–202, Washington, DC, USA, 2003. IEEE Computer Society. 1, 5, 6

[29] A. Seki and M. Pollefeys. Patch based confidence prediction for dense disparity map. In *British Machine Vision Conference (BMVC)*, 2016. 2, 3, 4, 5, 6, 7, 8

[30] X. Song, X. Zhao, H. Hu, and L. Fang. Edgestereo: A context integrated residual pyramid network for stereo matching. In *Asian Conference on Computer Vision (ACCV)*, 2018. 3

[31] A. Spyropoulos, N. Komodakis, and P. Mordohai. Learning to detect ground control points for improving the accuracy of stereo matching. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1621–1628. IEEE, 2014. 2, 3, 4

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

[33] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Proceedings of the Sixth International Conference on Computer Vision*, ICCV '98, pages 839–, Washington, DC, USA, 1998. IEEE Computer Society. 3

[34] A. Tonioni, M. Poggi, S. Mattoccia, and L. Di Stefano. Unsupervised adaptation for deep stereo. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 3

[35] A. Tonioni, F. Tosi, M. Poggi, S. Mattoccia, and L. Di Stefano. Real-time self-adaptive deep stereo. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3

[36] 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. 2, 6

[37] F. Tosi, M. Poggi, S. Mattoccia, A. Tonioni, and L. di Stefano. Learning confidence measures in the wild. In *British Machine Vision Conference 2017, BMVC 2017, London, UK, September 4-7, 2017*, 2017. 2

[38] G. Yang, H. Zhao, J. Shi, Z. Deng, and J. Jia. Segstereo: Exploiting semantic information for disparity estimation. In *15th European Conference on Computer Vision (ECCV)*, 2018. 3

[39] L. Yu, Y. Wang, Y. Wu, and Y. Jia. Deep stereo matching with explicit cost aggregation sub-architecture, 2018. 3

[40] Y. W. J. S. Z. Ma, K. He and E. Wu. Constant time weighted median filtering for stereo matching and beyond. In *International Conference on Computer Vision*, ICCV, 2013. 3, 5, 6

[41] 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)*, ECCV '94, pages 151–158, Secaucus, NJ, USA, 1994. Springer-Verlag New York, Inc. 5

[42] J. Zbontar and Y. LeCun. Computing the stereo matching cost with a convolutional neural network. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 3

[43] J. Zbontar and Y. LeCun. Stereo matching by training a convolutional neural network to compare image patches. *Journal of Machine Learning Research*, 17:1–32, 2016. 1, 3, 5, 6, 8

[44] X. L. Zhang Q. and J. J. 100+ times faster weighted median filter. In *IEEE Conference on Computer Vision and Pattern Recognition*, CVPR, 2014. 3, 5, 6, 8