SMD-Nets: Stereo Mixture Density Networks

SMD-Nets: Stereo Mixture Density Networks

Fabio Tosi[1*]   Yiyi Liao[2]   Carolin Schmitt[2]   Andreas Geiger[2]

[1]Department of Computer Science and Engineering (DISI), University of Bologna
[2] Autonomous Vision Group, MPI-IS / University of Tübingen

[1]{fabio.tosi5}@unibo.it [2]{firstname.lastname}@tue.mpg.de

Abstract

Despite stereo matching accuracy has greatly improved by deep learning in the last few years, recovering sharp boundaries and high-resolution outputs efficiently remains challenging. In this paper, we propose Stereo Mixture Density Networks (SMD-Nets), a simple yet effective learning framework compatible with a wide class of 2D and 3D architectures which ameliorates both issues. Specifically, we exploit bimodal mixture densities as output representation and show that this allows for sharp and precise disparity estimates near discontinuities while explicitly modeling the aleatoric uncertainty inherent in the observations. Moreover, we formulate disparity estimation as a continuous problem in the image domain, allowing our model to query disparities at arbitrary spatial precision. We carry out comprehensive experiments on a new high-resolution and highly realistic synthetic stereo dataset, consisting of stereo pairs at 8Mpx resolution, as well as on real-world stereo datasets. Our experiments demonstrate increased depth accuracy near object boundaries and prediction of ultra high-resolution disparity maps on standard GPUs. We demonstrate the flexibility of our technique by improving the performance of a variety of stereo backbones.

1. Introduction

Stereo matching is a long standing and active research topic in computer vision. It aims at recovering dense correspondences between image pairs by estimating the *disparity* between matching pixels, required to infer depth through triangulation. It also plays a crucial role in many areas like 3D mapping, scene understanding and robotics.

Traditional stereo matching algorithms apply handcrafted matching costs and engineered regularization strategies. More recently, learning methods based on Convolutional Neural Networks (CNNs) have proven to be superior, given the increasing availability of large stereo datasets

---

*Work done as intern at MPI-IS.
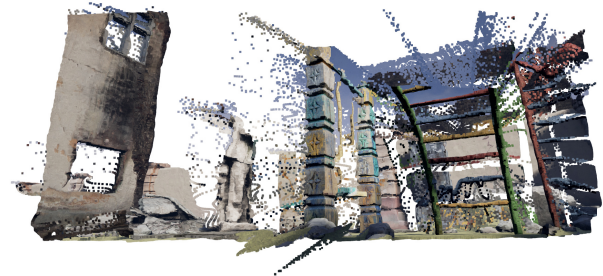
(a) PSM [4]
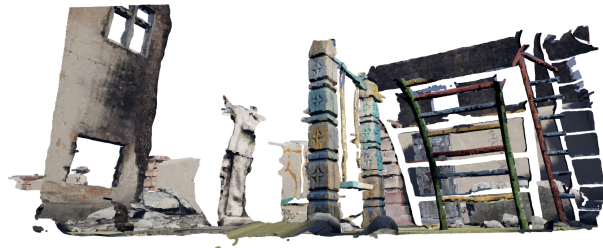
(b) PSM [4] + Ours

Figure 1: **Point Cloud Comparison** between the stereo network PSM [4] and our Stereo Mixture Density Network (SMD-Net) on the UnrealStereo4K dataset. Notice how SMD-Net notably alleviates bleeding artifacts near object boundaries, resulting in more accurate 3D reconstructions.

[11, 10, 52]. Although such methods produce compelling results, two major issues remain unsolved: predicting accurate depth boundaries and generating high-resolution outputs with limited memory and computation.

The first issue is shown in Fig. 1a: As neural networks are smooth function approximators, they often poorly reconstruct object boundaries, causing "bleeding" artifacts (i.e., flying pixels) when converted to point clouds. These artifacts can be detrimental to subsequent applications such as 3D reconstruction or 3D object detection. Thus, while being ignored by most commonly employed disparity metrics, accurate 3D reconstruction of contours is a desirable property for any stereo matching algorithm.

Furthermore, existing methods are limited to discrete predictions at pixel locations of a fixed resolution image grid, while geometry is a piecewise continuous quantity where object boundaries may not align with pixel centers. Increasing the output resolution by adding extra upsampling layers partially addresses this problem as this leads to a significant increase in memory and computation.

In this work, we address both issues. Our key contribution is to learn a representation that is precise at object boundaries and scales to high output resolutions. In particular, we formulate the task as a continuous estimation problem and exploit bimodal mixture densities [2] as output representation. Our simple formulation lets us (1) avoid bleeding artifacts at depth discontinuities, (2) regress disparity values at arbitrary spatial resolution with constant memory and (3) provides a measure for aleatoric uncertainty.

We illustrate the boundary bleeding problem and our solution to it in Fig. 2. While classical deep networks for stereo regression suffer from smoothness bias and are incapable of representing sharp disparity discontinuities, the proposed Stereo Mixture Density Networks (SMD-Nets) effectively address this issue. The key idea is to alter the output representation adopting a mixture distribution such that sharp discontinuities can be regressed *despite* the fact that the underlying neural networks are only able to make smooth predictions (note that all curves in Fig. 2b are indeed smooth while the predicted disparity is discontinuous).

Furthermore, the proposed model is capable of regressing disparity values at arbitrary continuous locations in the image, effectively solving a stereo super-resolution task. In combination with the proposed representation, this allows for regressing sharp discontinuities at sub-pixel resolution while keeping memory requirements constant.

In summary, we present: (i) A novel learning framework for stereo matching that exploits compactly parameterized bimodal mixture densities as output representation and can be trained using a simple likelihood-based loss function. (ii) A continuous function formulation aimed at estimating disparities at arbitrary spatial resolution with constant memory footprint. (iii) A new large-scale synthetic binocular stereo dataset with ground truth disparities at 3840×2160 resolution, comprising photo-realistic renderings of indoor and outdoor environments. (iv) Extensive experiments on several datasets demonstrating improved accuracy at depth discontinuities for various backbones on binocular stereo, monocular and active depth estimation tasks.

Our source code and dataset are available at https://github.com/fabiotosi92/SMD-Nets.

2. Related Work

Deep Stereo Matching: Stereo has a long history in computer vision [41]. With the rise of deep learning, CNN based methods for stereo were pioneered in [56] with the aim of replacing the traditional matching cost computation.

More recent works attempt to solve the stereo matching task without hand-crafted post processing steps. They can be categorized into 2D architectures and 3D architectures. In the first category, [22, 32, 15, 53, 54, 46, 1] extend the seminal DispNet [24], an end-to-end network for disparity regression. The second class, instead, consists of architectures that explicitly construct 3D feature cost volumes by means of concatenation/feature difference [4, 18, 47, 48, 8, 57, 49, 55, 29, 3, 7, 45, 50, 21] and group-wise correlation [14]. A thorough review of these works can be found in [36]. We stress once again how such networks, although achieving state-of-the-art results on most stereo benchmarks, suffer from severe over-smoothing at object discontinuities which are not captured by commonly employed disparity metrics, but which matter for many downstream applications. Therefore, the ideas proposed in this work to address this issue are orthogonal to the aforementioned networks and can be advantageously combined with nearly any stereo backbone.

Disparity Output Representation: Standard stereo networks directly regress a scalar disparity at every pixel. This output representation suffers from over-smoothing and does not expose the underlying aleatoric uncertainty. The latter problem can be addressed by modeling the disparity using a *parametric* distribution, e.g., a Gaussian or Laplacian distribution [16, 25] while the over-smoothing issue remains unsolved. A key result of our work is to demonstrate that replacing the unimodal output representation with a bimodal one is sufficient to significantly alleviate this problem.

Another line of methods estimate a *non-parametric* distribution over a set of discrete disparity values. However, this approach leads to inaccurate results when the estimated distribution is multi-modal [17]. Some works tackle the problem by enforcing a unimodal constraint during training [5, 58]. In contrast, we explicitly model the bimodal nature of the distribution at object boundaries by adopting a simple and effective bimodal representation. In concurrent work, [9] also predicts multi-modal distributions supervised by a heuristically designed multi-modal ground truth over a set of depth values. In contrast to them, our bimodal approach can be learned by maximizing the likelihood without requiring direct supervision on the distribution itself.

Continuous Function Representation: Existing deep stereo networks use fully convolutional neural networks and make predictions at discrete pixel locations. Recently, continuous function representations have gained attention in many areas, including 3D reconstruction [27, 33, 6, 39, 44, 30, 35], texture estimation [31], image synthesis [28, 42] and semantic segmentation [20]. To the best of our knowledge, we are the first to adopt a continuous function representation for disparity estimation, allowing us to predict a
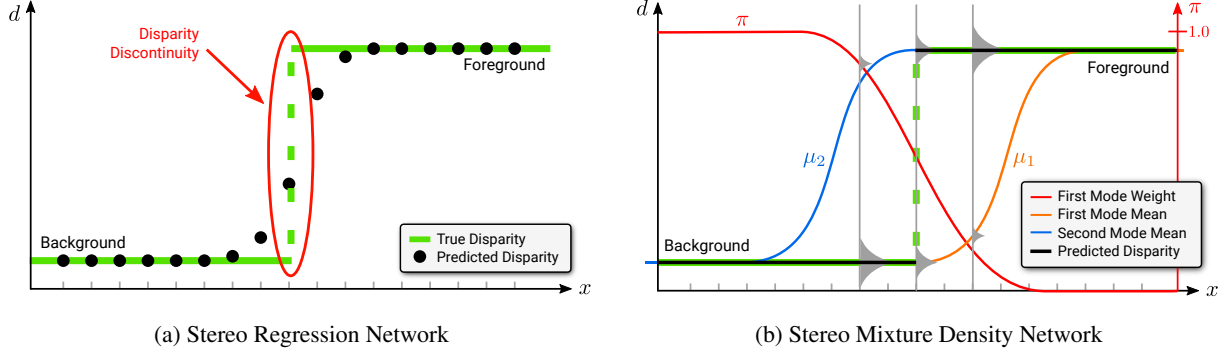
(a) Stereo Regression Network   (b) Stereo Mixture Density Network

Figure 2: **Overcoming the Smoothness Bias with Mixture Density Networks.** For clarity, we visualize the disparity d only for a single image row. (a) Classical deep networks for stereo regression suffer from smoothness bias and hence continuously interpolate object boundaries. In addition, disparity values are typically predicted at discrete spatial locations. (b) In this work, we propose to use a bimodal Laplacian mixture distribution (illustrated in gray) with weight π as output representation which can be queried at any continuous spatial location x. This allows our model to accurately capture uncertainty close to depth discontinuities while at inference time recovering sharp edges by selecting the mode with the highest probability density. In this example, the first mode (μ_1, b_1) models the background and the second mode (μ_2, b_2) models the foreground disparity close to the discontinuity. When the probability density of the foreground mode becomes larger than the probability density of the background mode, the most likely disparity sharply transitions from the background to the foreground value.

disparity value at any continuous pixel location. In contrast to works that allow for high-resolution stereo matching by designing memory efficient architectures [51, 13], our simple output representation is able to exploit ground truth disparity maps at a higher resolution than the input stereo pair, thus effectively learning stereo super-resolution.

3. Method

Fig. 3 illustrates our model. We first encode a stereo pair into a feature map using a convolutional backbone (left). Next, we estimate parameters of a mixture density distribution at any continuous 2D location via a multi-layer perceptron head, taking the bilinearly interpolated feature vector as input (middle). From this, we obtain a disparity as well as uncertainty map (right). We now explain our model, loss function and training protocol in detail.

3.1. Problem Statement

Let $\mathbf{I} \in \mathbb{R}^{W \times H \times 6}$ denote an RGB stereo pair for which we aim to predict a disparity map $\mathbf{D}$ at arbitrary resolution. As shown in Fig. 2, classical stereo regression networks suffer from over-smoothing due to the smoothness bias of neural networks. In this work, we exploit a mixture distribution as output representation [2] to overcome this limitation.

More specifically, we propose to use a bimodal Laplacian mixture distribution with weight π and two modes (μ_1, b_1), (μ_2, b_2) to model the continuous probability distribution over disparities at a particular pixel. Using two modes allows our model to capture both the foreground as well as the background disparity at object boundaries. At inference time, we recover sharp object boundaries by se-

lecting the mode with the highest density value. Thus, our model is able to transition from one disparity to another in a discontinuous fashion while at the same time relying only on the regression of functions ($\pi, \mu_1, b_1, \mu_2, b_2$) which are smooth with respect to the image domain and which therefore can easily be represented using neural networks.

3.2. Stereo Mixture Density Networks

We now formally describe our model. Let

$$\Psi_\theta : \mathbb{R}^{W \times H \times 6} \to \mathbb{R}^{W \times H \times D} \qquad (1)$$

denote a *stereo backbone* network with parameters θ as shown in Fig. 3 (left). Ψ_θ takes as input the stereo pair $\mathbf{I}$ and outputs a D-dimensional feature map, represented in the domain of the reference image (e.g. the left image of a stereo pair). Examples for such networks are standard 2D convolutional networks, or networks which perform 3D convolutions. For the 2D networks, the stereo pair can be concatenated as input or processed by means of siamese towers with shared weights as typically done for 3D architectures. Similarly, this generic formulation also applies to the structured light setting (e.g., Kinect setting where $\mathbf{I} \in \mathbb{R}^{W \times H}$) and the monocular depth estimation problem ($\mathbf{I} \in \mathbb{R}^{W \times H \times 3}$).

As geometry is a piecewise continuous quantity, we apply a deterministic transformation to obtain feature points for any continuous location in $\mathbb{R}^{W \times H}$. More specifically, for every continuous 2D location $\mathbf{x} \in \mathbb{R}^2$, we bilinearly interpolate the features from its four nearest pixel locations in the feature map $\mathbb{R}^{W \times H \times D}$. More formally, we describe this transformation as:

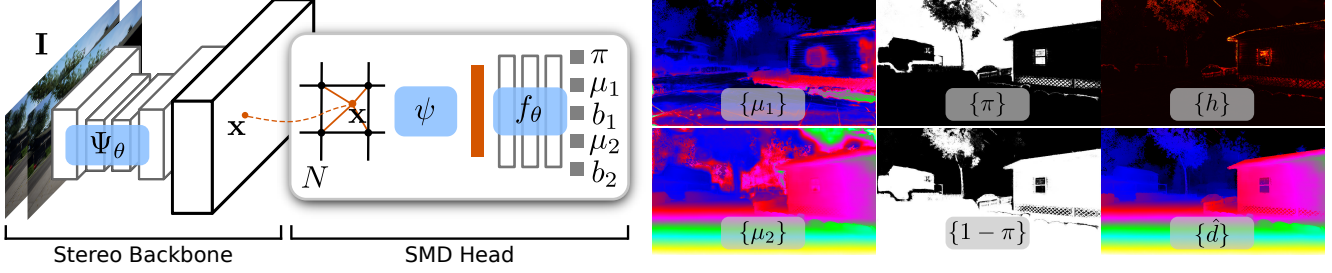$$\psi : \mathbb{R}^2 \times \mathbb{R}^{W \times H \times D} \to \mathbb{R}^D \qquad (2)$$

Figure 3: **Method Overview.** We assume a 2D or 3D stereo backbone network Ψ_θ which takes as input a stereo pair $\mathbf{I}$ (either concatenated or processed by siamese towers), and outputs a D-dimensional feature map in the domain of the reference image. Given any continuous 2D location $\mathbf{x}$, we query its feature from the feature map via bilinear interpolation as denoted by ψ. The interpolated feature vector is then fed into a multi-layer perceptron f_θ to estimate a five-dimensional vector $(\pi, \mu_1, b_1, \mu_2, b_2)$ which represents the parameters of a bimodal distribution. N denotes the number of points randomly sampled at continuous 2D locations during training and the number of pixels during inference. On the right we show maps of $\mu_1, \mu_2, \pi, 1 - \pi$, uncertainty h and predicted disparity $\hat{d}$.

Finally, we employ a multi-layer perceptron to map this abstract feature representation to a five-dimensional vector $(\pi, \mu_1, b_1, \mu_2, b_2)$ which represents the parameters of a univariate bimodal mixture distribution:

$$f_\theta : \mathbb{R}^D \to \mathbb{R}^5 \qquad (3)$$

Note that we have re-used the parameter symbol θ to simplify notation. In the following, we use θ to denote all parameters of our model. We refer to $f_\theta(\psi(\cdot, \cdot))$ as *SMD Head*, see Fig. 3 for an illustration.

To robustly model a distribution over disparities which can express two modes close to disparity discontinuities, we choose a bimodal Laplacian mixture as output representation:

$$p(d) = \frac{\pi}{2\,b_1} e^{-\frac{|d - \mu_1|}{b_1}} + \frac{1 - \pi}{2\,b_2} e^{-\frac{|d - \mu_2|}{b_2}} \qquad (4)$$

In summary, our model can be compactly expressed as:

$$p(d|\mathbf{x}, \mathbf{I}, \theta) = p(d|f_\theta(\psi(\mathbf{x}, \Psi_\theta(\mathbf{I})))) \qquad (5)$$

At inference time, we determine the final disparity $\hat{d}$ by choosing the mode with the highest density value:

$$\hat{d} = \underset{d \in \{\mu_1, \mu_2\}}{\operatorname{argmax}} \; p(d) \qquad (6)$$

Note that our formulation allows to query the disparity $\hat{d} \in \mathbb{R}$ at any continuous 2D pixel location, enabling ultra high-resolution predictions with sharply delineated object boundaries. This is illustrated in Fig. 4.

Our model also allows for capturing the aleatoric uncertainty of the predicted disparity by evaluating the differential entropy of the continuous mixture distribution as:

$$h = - \int p(d) \log p(d) \, \mathrm{d}d \qquad (7)$$

In practice, we use numerical quadrature to obtain an approximation of the integral.

3.3. Loss Function

We consider the supervised setting and train our model by minimizing the negative log-likelihood loss:

$$\mathcal{L}_{NLL}(\theta) = -\mathbb{E}_{d, \mathbf{x}, \mathbf{I}} \log p(d|\mathbf{x}, \mathbf{I}, \theta) \qquad (8)$$

where the input $\mathbf{I}$ is randomly sampled from the dataset, $\mathbf{x}$ is a random pixel location in the continuous image domain $\Omega = [0, W - 1] \times [0, H - 1]$, sampled as described in Sec. 3.4, and d is the ground truth disparity at location $\mathbf{x}$.

3.4. Training Protocol

Sampling Strategy: While a naïve strategy samples pixel locations $\mathbf{x}$ randomly and uniformly from the image domain Ω, our framework also allows for exploiting custom sampling strategies to focus on depth discontinuities during training. We adopt a *Depth Discontinuity Aware (DDA)* sampling approach during training that explicitly favors points located near object boundaries while at the same time maintaining a uniform coverage on the entire image space. More specifically, given a ground truth disparity map at training time, we first compute an object boundary mask in which a pixel is considered to be part of the boundary if its (4-connected) neighbors have a disparity that differs by more than 1 from its own disparity. This mask is then dilated using a $\rho \times \rho$ kernel to enlarge the boundary region. We report an analysis using different ρ values in the experimental section. Given the total number of training points N, we randomly and uniformly select $N/2$ points from the domain of all pixels belonging to depth discontinuity regions and $N/2$ points uniformly from the continuous domain of all remaining pixels. At inference time, we leverage our model to predict disparity values at each location of an (arbitrary resolution) grid.

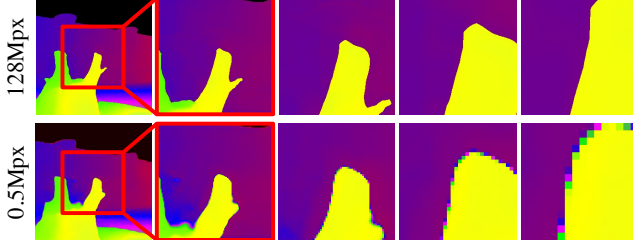**Stereo Super-Resolution:** Our continuous formulation al-

Figure 4: **Ultra High-resolution Estimation.** Comparison of our model using the PSM backbone at 128Mpx resolution (top) to the original PSM at 0.5Mpx resolution (bottom), both taking stereo pairs at 0.5Mpx resolution as input. Each column shows a different zoom-level. Note how our method leads to sharper boundaries and high resolution outputs.

lows us to exploit ground truth at higher resolution than the input $\mathbf{I}$, which we refer to as stereo super-resolution. In contrast, classical stereo methods cannot realize arbitrary super-resolution without changing their architecture.

4. Experimental Results

In this section, we first describe the datasets used for evaluation and implementation details. We then present an extensive evaluation that demonstrates the benefits of the proposed SMD Head in combination with different stereo backbones on several distinct tasks.

4.1. Datasets

UnrealStereo4K: Motivated by the lack of large-scale, realistic and high-resolution stereo datasets, we introduce a new photo-realistic *binocular stereo* dataset at 3840×2160 resolution with pixel-accurate ground truth. We create this synthetic dataset using the popular game engine Unreal Engine combined with the open-source plugin UnrealCV [37]. We additionally create a synthetic *active monocular* dataset (mimicking the Kinect setup) at 4112×3008 resolution by warping a gray-scale reference dot pattern to each image, following [38]. We split the dataset into 7720 training pairs, 80 validation pairs and 200 *in-domain* test pairs. To evaluate the generalization ability of our method, we also create an *out-of-domain* test set by rendering 200 stereo pairs from an unseen scene. Similarly, the active dataset contains 3856 training images, 40 validation images, 100 test images.

RealActive4K: We further collect a small real-world active dataset of an indoor room with a Kinect-like stereo sensor, including 2570 images at a resolution 4112×3008 pixels from which we use 2500 for training, 20 for validation and 50 for testing. We perform Block Matching with left-right consistency check to use as co-supervision for training models jointly on synthetic (UnrealStereo4K) and real data.

KITTI 2015 [26]: The KITTI dataset is a collection of real-world stereo images depicting driving scenarios. It contains 200 training pairs with sparse ground truth depth maps collected by a LiDAR and 200 testing pairs. We divide the KITTI training set into 160 training stereo pairs and 40 validation stereo pairs, following [45].

Middlebury v3 [40]: Middlebury v3 is a small high-resolution stereo dataset depicting indoor scenes under controlled lighting conditions containing 10 training pairs and 10 testing pairs with dense ground truth disparities.

4.2. Implementation Details

Architecture: In principle, our SMD Head is compatible with any stereo backbone ψ_θ from the literature. In our implementation, we build on top of two state-of-the-art 3D stereo architectures: Pyramid Stereo Matching (PSM) network [4] and Hierarchical Stereo Matching (HSM) network [51]. PSM is a well-known and popular stereo network while HSM represents a method with good trade-off between accuracy and computation. Moreover, we also adopt a naïve U-Net structure [12] that takes as input concatenated images of a stereo pair in order to show the effectiveness of our model on 2D architectures. For the aforementioned networks, we follow the official code provided by the authors.

Our SMD Head f_θ is implemented as a multi-layer perceptron (MLP) following [39]. More specifically, the number of neurons is $(D, 1024, 512, 256, 128, 5)$. We use sine activations [44] except for the last layer that uses a sigmoid activation for regressing the five-parameter output. For the 3D backbone, we select the matching probabilities from the cost volume in combination with features of Ψ_θ at different resolutions as input to our SMD Head. For the 2D backbone case, instead, we select features from different layers of the decoder. We refer the reader to the supplementary material for details.

Training: We implement our approach in PyTorch [34] and use Adam with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ as optimizer [19]. We train all models from scratch using a single NVIDIA V100 GPU. During training, we use random crops from $\mathbf{I}$ as input to the stereo backbone and sample $N = 50,000$ training points from each crop. We scale the ground truth disparity to $[0, 1]$ for each dataset for numerical stability. Moreover, for RGB inputs we perform chromatic augmentations on the fly, including random brightness, gamma and color shifts sampled from uniform distributions. We further apply horizontal and vertical random flipping while adapting the ground truth disparities accordingly. Please see the supplementary material for details regarding the training procedure for each dataset.

Evaluation Metrics: Following [5], we evaluate the *Soft Edge Error (SEE_k)* metric on pixels belonging to object

boundaries, defined as the minimum absolute error between the predicted disparity and the corresponding local ground truth patch of size $k \times k$ ($k = \{3, 5\}$ in our experiments). Intuitively, *SEE* penalizes over-smoothing artifacts stronger compared to small misalignments in a local window, where the former is more harmful to subsequent applications.

While not our main focus, we also report the *End Point Error (EPE)* as the standard error metric obtained by averaging the absolute difference between predictions and ground truth disparity values to evaluate the overall performance. For both *SEE* and *EPE*, we compute the average (Avg) and $\sigma(\Delta)$ metrics, with the latter one representing the percentage of pixels having errors greater than Δ.

4.3. Ablation Study

We first examine the impact of different components and training choices of the proposed SMD-Nets on the *in-domain* UnrealStereo4K test set. Unless specified otherwise, we use 960×540 as resolution for the binocular input **I** and 3840×2160 for the corresponding ground truth, used for both supervision and testing purposes. The active input images consist of random dot patterns where the dots become indistinguishable at low resolution (e.g., 960×540). Therefore we use 2056×1504 as active input size while keeping the ground truth dimension at 4112×3008.

Output Representation: In Tab. 1, we evaluate the effectiveness of our mixture density output representation across both, 2D and 3D stereo backbones on multiple tasks including binocular stereo, monocular depth and active depth. We adopt U-Net and PSM on the binocular stereo dataset as representatives of 2D and 3D backbones and report results of HSM in the supplementary for the sake of space. We also use the same U-Net backbone for a monocular depth estimation task by replacing the input with only the reference image of a binocular stereo pair to show the advantage of our method on various tasks. For the active setup, we choose HSM as it represents a network designed specifically for high-resolution inputs which takes as input the monocular active image and the *fixed* reference dot pattern.

We compare our bimodal distribution to two other output representations, standard disparity regression and a unimodal Laplacian distribution [16]. For fairness, we implement these baselines by replacing the last layer of our SMD Head to predict the disparity d or the unimodal parameters (μ, b), respectively, where the former is trained with a standard L1 loss while the latter with a negative log-likelihood loss. For all cases we use the proposed bilinear feature interpolation and the naïve random sampling strategy.

Tab. 1 shows that the proposed method effectively addresses the over-smoothing problem at object boundaries, achieving the lowest *SEE* for all backbones on all tasks, compared to both the standard disparity regression and the unimodal representation. Moreover, we observe that the

	Ψ_θ	Dim.	SEE₃ Avg	$\sigma(1)$	$\sigma(2)$	SEE₅ Avg	$\sigma(1)$	$\sigma(2)$	EPE Avg	$\sigma(3)$
Binocular Stereo	U-Net	1	2.15	41.69	24.16	2.03	39.65	22.98	1.48	8.18
	(2D)	2	2.38	42.28	25.74	2.26	40.42	24.57	1.97	10.44
	[12]	5	**1.57**	**30.06**	**14.77**	**1.45**	**28.05**	16.57	**1.28**	**5.94**
	PSM	1	1.98	36.32	20.35	1.85	34.42	19.21	**1.10**	5.52
	(3D)	2	2.50	39.40	23.63	2.37	37.57	22.51	1.88	7.73
	[4]	5	**1.52**	**26.98**	**12.68**	**1.38**	**24.93**	**11.49**	1.11	**4.80**
Mono.	U-Net	1	3.29	60.18	41.37	3.25	58.49	40.08	**4.21**	35.92
	(2D)	2	4.01	61.06	43.19	3.86	59.40	41.90	5.49	41.88
	[12]	5	**2.92**	**51.32**	**32.33**	**2.78**	**49.54**	**31.06**	4.06	**30.59**
Active	HSM	1	3.40	47.87	24.80	3.18	46.14	23.76	**1.29**	5.84
	(3D)	2	4.93	57.05	33.44	4.69	55.47	32.41	2.83	10.70
	[51]	5	**2.69**	**41.84**	**17.35**	**2.43**	**39.83**	**16.17**	1.42	**5.48**

Table 1: **Output Representation** analysis on the UnrealStereo4K test set. "Dim." refers to the output dimension of the SMD Head where 1 indicates the point estimate d, 2 the unimodal output representation (μ, b) [16] and 5 our bimodal formulation $(\pi, \mu_1, b_1, \mu_2, b_2)$.

Sampling	ρ	SEE₃ Avg	$\sigma(1)$	$\sigma(2)$	SEE₅ Avg	$\sigma(1)$	$\sigma(2)$	EPE Avg	$\sigma(3)$
Random	-	1.52	26.98	12.68	1.38	24.93	11.49	1.11	4.80
DDA	0	1.34	21.62	9.77	1.19	19.58	8.59	1.08	4.44
DDA	10	**1.13**	**18.64**	**8.69**	**0.98**	**16.67**	**7.55**	**0.92**	**3.88**
DDA	20	1.30	20.42	9.88	1.15	18.40	8.71	1.11	4.44

Table 2: **Sampling Strategy** analysis on the UnrealStereo4K test set using the PSM backbone.

Eval. GT	Training GT	SEE₃ Avg	$\sigma(1)$	$\sigma(2)$	SEE₅ Avg	$\sigma(1)$	$\sigma(2)$	EPE Avg	$\sigma(3)$
960×540	960×540	1.19	20.36	9.16	0.93	16.55	7.08	1.02	4.30
960×540	3840×2160	**0.98**	**15.42**	**7.05**	**0.78**	**12.44**	**5.54**	**0.89**	**3.81**
3840×2160	960×540	1.33	23.35	10.82	1.19	21.34	9.63	1.03	4.30
3840×2160	3840×2160	**1.13**	**18.64**	**8.69**	**0.98**	**16.67**	**7.55**	**0.92**	**3.88**

Table 3: **Ground Truth Resolution** analysis on the UnrealStereo4K test set using the PSM backbone.

unimodal representation sacrifices *EPE* for capturing the uncertainty, while our method is on par with the standard L1 regression. On the stereo dataset, the 3D backbone (PSM) consistently outperforms the 2D backbone (U-Net), therefore we use PSM for the following ablation experiments.

Sampling Strategy: In Tab. 2, we show the impact of the sampling strategy adopted during training. More specifically, we compare the naïve uniform sampling strategy and the proposed *DDA* approach using different dilation kernel sizes $\rho \times \rho$. As can be observed, *DDA* enables SMD-Nets to focus on depth discontinuities, resulting in better *SEE* compared to random point selection. Moreover, we observe that sampling exactly at depth boundaries (i.e., $\rho = 0$) leads to slightly degraded *EPE* and is less effective on *SEE* which penalizes small misalignment in a local window. Instead, setting $\rho = 10$ allows the network to focus on larger regions near edges and results in the best performance, while increasing ρ does not improve performance further. Finally, it is worth to notice that this strategy also allows our model to improve the overall performance, achieving lower *EPE* metrics. In the following experiments, we thus adopt the *DDA* strategy using $\rho = 10$ for our SMD-Nets.

Method	In-domain										Out-of-domain									
	SEE_3			SEE_5			EPE				SEE_3			SEE_5			EPE			
	Avg	$\sigma(1)$	$\sigma(2)$	Avg	$\sigma(1)$	$\sigma(2)$	Avg	$\sigma(1)$	$\sigma(2)$	$\sigma(3)$	Avg	$\sigma(1)$	$\sigma(3)$	Avg	$\sigma(1)$	$\sigma(3)$	Avg	$\sigma(1)$	$\sigma(2)$	$\sigma(3)$
PSM [4]	1.73	33.06	16.57	1.61	31.11	15.44	1.09	11.88	6.94	5.19	2.19	36.94	20.07	1.99	34.09	18.25	1.53	16.92	10.25	7.83
PSM [4] + BF [23]	1.65	30.93	15.26	1.52	28.92	14.10	1.10	11.81	6.95	5.23	2.16	35.64	19.16	1.95	32.76	17.32	1.56	18.89	10.28	7.89
PSM [4] + SM [5]	1.50	29.22	12.71	1.37	27.16	11.54	1.10	11.65	6.69	4.97	2.03	33.91	16.74	1.82	30.92	14.82	1.54	16.43	9.73	7.36
PSM [4] + CE + SM [5]	1.33	27.31	10.14	1.19	25.25	8.99	**0.86**	10.40	**4.93**	3.50	1.84	29.87	13.30	1.62	26.84	11.46	1.37	13.29	7.84	**6.03**
PSM [4] + Ours	**1.13**	**18.64**	**8.69**	**0.98**	**16.67**	**7.55**	0.92	**8.24**	5.06	3.88	**1.59**	**24.58**	12.54	**1.38**	**21.63**	10.73	**1.27**	**12.11**	7.69	6.06
HSM [51]	2.01	41.63	23.81	1.89	39.69	22.62	1.16	14.81	8.20	5.84	2.43	44.49	26.17	2.24	41.74	24.33	1.75	22.03	12.73	9.23
HSM [51] + BF [23]	1.88	39.68	21.70	1.77	37.67	20.49	1.19	14.78	8.21	5.88	2.39	43.60	24.14	2.19	40.82	23.28	1.80	22.05	12.79	9.33
HSM [51] + SM [5]	1.83	40.52	22.30	1.70	38.53	21.07	1.17	14.73	8.11	5.74	2.31	43.76	25.16	2.11	40.97	23.29	1.76	21.88	12.54	9.03
HSM [51] + CE + SM [5]	2.00	45.71	25.99	1.87	43.72	24.71	1.17	16.17	8.12	5.46	2.61	48.27	28.84	2.41	45.56	26.98	1.91	26.12	14.40	10.14
HSM [51] + Ours	**1.31**	**24.31**	**10.81**	**1.17**	**22.30**	**9.67**	**1.00**	**11.40**	**6.09**	**4.34**	**2.03**	**34.82**	**17.75**	**1.82**	**31.88**	**15.83**	**1.66**	**19.16**	**10.72**	**7.77**

Table 4: **Comparison on UnrealStereo4K.** All methods evaluated on ground truth at 3840×2160 given input size 960×540.

(a) PSM [4] (b) PSM [4] + CE + SM [5] (c) PSM + Ours (d) GT, Input
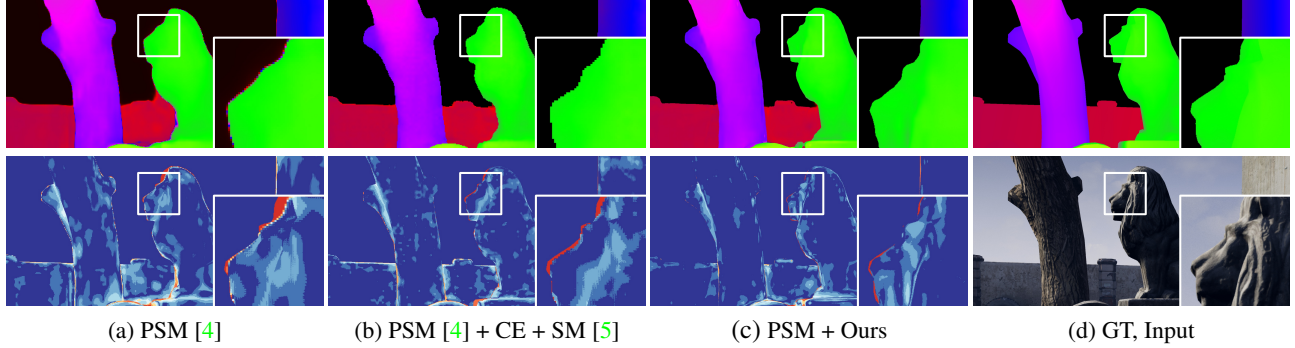
Figure 5: **Qualitative Results on UnrealStereo4K.** The first row shows the predicted disparity maps while the second row depicts the corresponding error maps. We zoom-in a patch in all images to better perceive details near depth boundaries.

Ground Truth Resolution: Tab. 3 shows the results of our model trained and tested on the stereo data using ground truth maps at different resolutions, while maintaining the input size at 960×540. Towards this goal, we train our model adopting ground truth disparities 1) resized to the same resolution as the input using nearest interpolation and 2) at the original resolution (i.e. 3840×2160). We notice that sampling points from higher resolution disparity maps always leads to better results compared to using low resolution ground truth. We remark that the proposed model effectively leverages high resolution ground truth thanks to its continuous formulation, without requiring additional memory compared to standard stereo networks based on CNNs.

4.4. Comparison to Existing Baselines

We now compare to several baselines [23, 5] which aim to address the over-smoothing problem. Bilateral median filtering (BF) is often adopted to sharpen disparity predictions [23, 43]. Chen et al. [5] address the over-smoothing problem of 3D stereo backbones using 1) a post-processing step to extract a single-modal (SM) distribution from the full discrete distribution; 2) a cross-entropy (CE) loss to enforce a unimodal distribution during training. We reimplement [5] as no official code is available. As [5] has been proposed for 3D backbones only, we use PSM [4] and HSM [51] as the stereo backbones in the following experiments.

UnrealStereo4K: Tab. 4 collects results obtained from dif-

ferent models on both *in-domain* and *out-of-domain* test splits of the binocular UnrealStereo4K dataset. We use the same input resolution of 960×540 for all methods. While our baseline methods can only use supervision with the same size as the input, we leverage our continuous formulation to supervise SMD-Nets using ground truth at 3840×2160, on which we also evaluate all methods. For our competitors, we upsample their outcome using nearest neighbor interpolation during testing. Both original PSM and HSM follow the same training setting of our SMD-Nets.

Tab. 4 suggests that BF [23] and SM [5] slightly improve *SEE* on both backbones while leading to degraded performance on *EPE* metrics. Using the CE loss combined with SM [5] leads to effective improvement on both *SEE* and *EPE* on the PSM backbone. Interestingly, we notice that adopting the same CE + SM strategy leads to worse performance on HSM. A possible explanation is that the CE loss requires to trilinearly interpolate a matching cost probability distribution to the full resolution $W \times H \times D_{max}$ (with D_{max} denoting the maximum disparity), where HSM predicts a less fine-grained cost distribution compared to PSM, thus making the cross-entropy loss less effective. Moreover, we remark that the CE loss is more expensive to compute, compared to our simple continuous likelihood-based formulation and CE + SM can only be applied on 3D backbones. In contrast, our approach based on the bimodal output representation notably outperforms our competitors on *SEE* on both the *in-domain* and *out-of-domain* test sets, showing

Method	SEE_3			SEE_5			EPE	
	Avg	$\sigma(1)$	$\sigma(2)$	Avg	$\sigma(1)$	$\sigma(2)$	Avg	$\sigma(3)$
PSM [4]	1.10	20.57	9.74	0.99	17.83	9.02	0.73	2.49
PSM [4] + CE + SM [5]	1.02	16.12	7.53	0.90	13.80	6.94	0.66	2.09
PSM [4] + Ours	**0.90**	**13.09**	**6.66**	**0.79**	**10.93**	**6.01**	**0.59**	**1.95**

Table 5: **Comparison on KITTI 2015 Validation Set** using boundaries extracted from instance segmentation masks to evaluate on depth discontinuity regions.

Method	All Areas			Non Occluded		
	Bg	Fg	All	Bg	Fg	All
GANet-deep [57]	1.48	3.46	1.81	1.34	3.11	1.63
HD³-Stereo [54]	1.70	3.63	2.02	1.56	3.43	1.87
GwcNet-g [14]	1.74	3.93	2.11	1.61	3.49	1.92
PSM [4]	1.86	4.62	2.31	1.71	4.31	2.14
PSM [4] + CE + SM [5]	**1.54**	4.33	2.14	1.70	3.90	1.93
PSM [4] + Ours	1.69	**4.01**	**2.08**	**1.54**	**3.70**	**1.89**

Table 6: **Comparison on KITTI 2015 Test Set**, evaluated on the official online benchmark. All the reported numbers represent official submissions from the authors.

| Method	SEE_3			SEE_5			EPE	
	Avg	$\sigma(1)$	$\sigma(2)$	Avg	$\sigma(1)$	$\sigma(2)$	Avg	$\sigma(3)$
PSM [4]	3.35	46.50	29.40	2.61	41.04	24.87	4.12	17.43
PSM [4] + CE + SM [5]	2.62	34.80	**19.02**	1.83	28.92	**14.11**	**2.80**	**12.12**
PSM [4] + Ours	**2.61**	**34.26**	19.83	**1.88**	**28.71**	15.32	3.03	13.60

Table 7: **Generalization on Middlebury v3**. All models are trained on UnrealStereo4K and evaluated on the training set of Middlebury v3 dataset.

how our strategy predicts better disparities near boundaries. Moreover, we highlight that we achieve consistently better estimates on standard *EPE* metrics compared to the original backbone while performing comparably to the CE + SM baseline. Fig. 5 shows our gains at object boundaries.

KITTI 2015: We fine-tune all methods trained using UnrealStereo4K on the KITTI 2015 training set. Since the provided ground truth disparities are sparse, we rely on the naïve random sampling strategy to train our model. On the validation set, we evaluate *SEE* on boundaries of instance segmentation maps from the KITTI dataset, following the evaluation procedure described in [5]. Furthermore, we predict disparities on the test set using the same fine-tuned model and submit to the online benchmark. Tab. 5 and Tab. 6 show our results using PSM as backbone (we provide additional results on the validation set adopting HSM in the supplement). Note that our SMD-Net not only achieves superior performance on both *SEE* and *EPE* metrics on the validation set compared to the original PSM and [5] (Tab. 5), but also outperforms both on the test set and is on par with state-of-the-art stereo networks on standard metrics of the KITTI benchmark (Tab. 6).

4.5. Synthetic-to-Real Generalization

Lastly, we demonstrate how models trained on the synthetic dataset generalize to the real-world domain for both binocular stereo and active depth estimation.
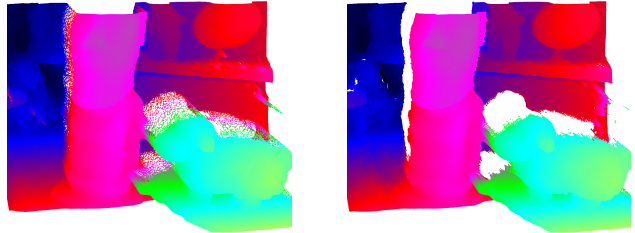
(a) Disparity Regression (L1)	(b) SMD Head (Bimodal)

Figure 6: **Generalization on RealActive4K** using the HSM backbone. The point clouds of standard disparity regression using L1 loss (a) show bleeding artifacts whereas our bimodal distribution (b) leads to clean reconstructions.

Middlebury v3: Tab. 7 reports the performance of supervised methods trained on the UnrealStereo4K and tested without fine-tuning on the training set of the Middlebury v3 dataset. We evaluate them using the high-resolution ground truth. Compared to the original PSM baseline, our SMD-Net achieves much better generalization on both *SEE* and *EPE* metrics while performing on par with [5].

RealActive4K: Moreover, we fine-tune our active depth models jointly on active UnrealStereo4K and RealActive4K with pseudo-ground truth from Block Matching. Fig. 6 shows that this allows for estimating sharp disparity edges for real captures even though Block Matching does not provide supervision in these areas. In contrast, standard disparity regression fails to predict clean object boundaries.

5. Conclusion

In this paper, we propose SMD-Nets, a novel stereo matching framework aimed at improving depth accuracy near object boundaries and suited for disparity super-resolution. By exploiting bimodal mixture densities as output representation combined with a continuous function formulation, our method is capable of predicting sharp and precise disparity values at arbitrary spatial resolution, notably alleviating the common over-smoothing problem in learning-based stereo networks. Our model is compatible with a broad spectrum of 2D and 3D stereo backbones. Our extensive experiments demonstrate the advantages of our strategy on a new high-resolution synthetic stereo dataset and on real-world stereo pairs. We plan to extend our bimodal output representation to other regression tasks such as optical flow and self-supervised depth estimation.

References

[1] Filippo Aleotti, Matteo Poggi, Fabio Tosi, and Stefano Mattoccia. Learning end-to-end scene flow by distilling single tasks knowledge. *Proc. of the Conf. on Artificial Intelligence (AAAI)*, 2020. 2

[2] Christopher M. Bishop. Mixture density networks. 1994. 2, 3

[3] Rohan Chabra, Julian Straub, Christopher Sweeney, Richard Newcombe, and Henry Fuchs. Stereodrnet: Dilated residual stereonet. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[4] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 2, 5, 6, 7, 8

[5] Chuangrong Chen, Xiaozhi Chen, and Hui Cheng. On the over-smoothing problem of cnn based disparity estimation. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2019. 2, 5, 7, 8

[6] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[7] Pier Luigi Dovesi, Matteo Poggi, Lorenzo Andraghetti, Miquel Martí, Hedvig Kjellström, Alessandro Pieropan, and Stefano Mattoccia. Real-time semantic stereo matching. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*, 2020. 2

[8] Shivam Duggal, Shenlong Wang, Wei-Chiu Ma, Rui Hu, and Raquel Urtasun. Deeppruner: Learning efficient stereo matching via differentiable patchmatch. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2019. 2

[9] Divyansh Garg, Yan Wang, B. Hariharan, M. Campbell, K. Weinberger, and Wei-Lun Chao. Wasserstein distances for stereo disparity estimation. *arXiv.org*, 2007.03085, 2020. 2

[10] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 32(11):1231–1237, 2013. 1

[11] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2012. 1

[12] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J. Brostow. Digging into self-supervised monocular depth prediction. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2019. 5, 6

[13] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[14] Xiaoyang Guo, Kai Yang, Wukui Yang, Xiaogang Wang, and Hongsheng Li. Group-wise correlation stereo network. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 8

[15] Eddy Ilg, Tonmoy Saikia, Margret Keuper, and Thomas Brox. Occlusions, motion and depth boundaries with a generic network for disparity, optical flow or scene flow estimation. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2018. 2

[16] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017. 2, 6

[17] Alex Kendall, Hayk Martirosyan, Saumitro Dasgupta, Peter Henry, Ryan Kennedy, Abraham Bachrach, and Adam Bry. End-to-end learning of geometry and context for deep stereo regression. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2017. 2

[18] Sameh Khamis, Sean Fanello, Christoph Rhemann, Adarsh Kowdle, Julien Valentin, and Shahram Izadi. Stereonet: Guided hierarchical refinement for real-time edge-aware depth prediction. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2018. 2

[19] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proc. of the International Conf. on Learning Representations (ICLR)*, 2015. 5

[20] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross B. Girshick. Pointrend: Image segmentation as rendering. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[21] Uday Kusupati, Shuo Cheng, Rui Chen, and Hao Su. Normal assisted stereo depth estimation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[22] Zhengfa Liang, Yiliu Feng, Yulan Guo, Hengzhu Liu, Wei Chen, Linbo Qiao, Li Zhou, and Jianfeng Zhang. Learning for disparity estimation through feature constancy. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[23] Ziyang Ma, Kaiming He, Yichen Wei, Jian Sun, and Enhua Wu. Constant time weighted median filtering for stereo matching and beyond. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, December 2013. 7

[24] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[25] Max Mehltretter and Christian Heipke. Cnn-based cost volume analysis as confidence measure for dense matching. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV) Workshops*, 2019. 2

[26] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015. 5

[27] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[28] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view syn-

thesis. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2020. 2

[29] Guang-Yu Nie, Ming-Ming Cheng, Yun Liu, Zhengfa Liang, Deng-Ping Fan, Yue Liu, and Yongtian Wang. Multi-level context ultra-aggregation for stereo matching. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[30] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[31] Michael Oechsle, Lars Mescheder, Michael Niemeyer, Thilo Strauss, and Andreas Geiger. Texture fields: Learning texture representations in function space. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2019. 2

[32] Jiahao Pang, Wenxiu Sun, Jimmy SJ Ren, Chengxi Yang, and Qiong Yan. Cascade residual learning: A two-stage convolutional neural network for stereo matching. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV) Workshops*, 2017. 2

[33] Jeong Joon Park, Peter Florence, Julian Straub, Richard A. Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[34] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 5

[35] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2020. 2

[36] Matteo Poggi, Fabio Tosi, Konstantinos Batsos, Philippos Mordohai, and Stefano Mattoccia. On the synergies between machine learning and stereo: a survey. *arXiv.org*, 2004.08566, 2020. 2

[37] Weichao Qiu, Fangwei Zhong, Yi Zhang, Siyuan Qiao, Zihao Xiao, Tae Soo Kim, Yizhou Wang, and Alan Yuille. Unrealcv: Virtual worlds for computer vision. *ACM Multimedia Open Source Software Competition*, 2017. 5

[38] Gernot Riegler, Yiyi Liao, Simon Donne, Vladlen Koltun, and Andreas Geiger. Connecting the dots: Learning representations for active monocular depth estimation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 5

[39] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2019. 2, 5

[40] Daniel Scharstein, Heiko Hirschmüller, York Kitajima, Greg Krathwohl, Nera Nešić, Xi Wang, and Porter Westling. High-resolution stereo datasets with subpixel-accurate

ground truth. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014. 5

[41] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision (IJCV)*, 47(1-3):7–42, 2002. 2

[42] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. GRAF: generative radiance fields for 3d-aware image synthesis. *Advances in Neural Information Processing Systems (NIPS)*, 2020. 2

[43] Meng-Li Shih, Shih-Yang Su, Johannes Kopf, and Jia-Bin Huang. 3d photography using context-aware layered depth inpainting. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020. 7

[44] Vincent Sitzmann, Julien N. P. Martel, Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. *Advances in Neural Information Processing Systems (NIPS)*, 2020. 2, 5

[45] Xiao Song, Xu Zhao, Liangji Fang, Hanwen Hu, and Yizhou Yu. Edgestereo: An effective multi-task learning network for stereo matching and edge detection. *International Journal of Computer Vision (IJCV)*, pages 1–21, 2020. 2, 5

[46] Alessio Tonioni, Fabio Tosi, Matteo Poggi, Stefano Mattoccia, and Luigi Di Stefano. Real-time self-adaptive deep stereo. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[47] Stepan Tulyakov, Anton Ivanov, and Francois Fleuret. Practical deep stereo (pds): Toward applications-friendly deep stereo matching. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 2

[48] Yan Wang, Zihang Lai, Gao Huang, Brian H Wang, Laurens Van Der Maaten, Mark Campbell, and Kilian Q Weinberger. Anytime stereo image depth estimation on mobile devices. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*, 2019. 2

[49] Zhenyao Wu, Xinyi Wu, Xiaoping Zhang, Song Wang, and Lili Ju. Semantic stereo matching with pyramid cost volumes. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2019. 2

[50] Haofei Xu and Juyong Zhang. Aanet: Adaptive aggregation network for efficient stereo matching. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[51] Gengshan Yang, Joshua Manela, Michael Happold, and Deva Ramanan. Hierarchical deep stereo matching on high-resolution images. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3, 5, 6, 7

[52] Guorun Yang, Xiao Song, Chaoqin Huang, Zhidong Deng, Jianping Shi, and Bolei Zhou. Drivingstereo: A large-scale dataset for stereo matching in autonomous driving scenarios. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1

[53] Guorun Yang, Hengshuang Zhao, Jianping Shi, Zhidong Deng, and Jiaya Jia. Segstereo: Exploiting semantic information for disparity estimation. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2018. 2

[54] Zhichao Yin, Trevor Darrell, and Fisher Yu. Hierarchical discrete distribution decomposition for match density estimation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 8

[55] Yurong You, Yan Wang, Wei-Lun Chao, Divyansh Garg, Geoff Pleiss, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar++: Accurate depth for 3d object detection in autonomous driving. *Proc. of the International Conf. on Learning Representations (ICLR)*, 2020. 2

[56] Jure Žbontar and Yann LeCun. Stereo matching by training a convolutional neural network to compare image patches. *Journal of Machine Learning Research (JMLR)*, 17(1):2287–2318, 2016. 2

[57] Feihu Zhang, Victor Prisacariu, Ruigang Yang, and Philip HS Torr. Ga-net: Guided aggregation net for end-to-end stereo matching. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 8

[58] Youmin Zhang, Yimin Chen, Xiao Bai, Suihanjin Yu, Kun Yu, Zhiwei Li, and Kuiyuan Yang. Adaptive unimodal cost volume filtering for deep stereo matching. *Proc. of the Conf. on Artificial Intelligence (AAAI)*, 2020. 2