# Stereo Risk: A Continuous Modeling Approach to Stereo Matching

Ce Liu [*] [1] [2] Suryansh Kumar [*] [3] Shuhang Gu [4] Radu Timofte [5] Yao Yao [1] Luc Van Gool [2] [6] [7]

## Abstract

We introduce Stereo Risk, a new deep-learning approach to solve the classical stereo-matching problem in computer vision. As it is well-known that stereo matching boils down to a per-pixel disparity estimation problem, the popular state-of-the-art stereo-matching approaches widely rely on regressing the scene disparity values, yet via discretization of scene disparity values. Such discretization often fails to capture the nuanced, continuous nature of scene depth. Stereo Risk departs from the conventional discretization approach by formulating the scene disparity as an optimal solution to a continuous risk minimization problem, hence the name "stereo risk". We demonstrate that $L^1$ minimization of the proposed continuous risk function enhances stereo-matching performance for deep networks, particularly for disparities with multi-modal probability distributions. Furthermore, to enable the end-to-end network training of the non-differentiable $L^1$ risk optimization, we exploited the implicit function theorem, ensuring a fully differentiable network. A comprehensive analysis demonstrates our method's theoretical soundness and superior performance over the state-of-the-art methods across various benchmark datasets, including KITTI 2012, KITTI 2015, ETH3D, SceneFlow, and Middlebury 2014.

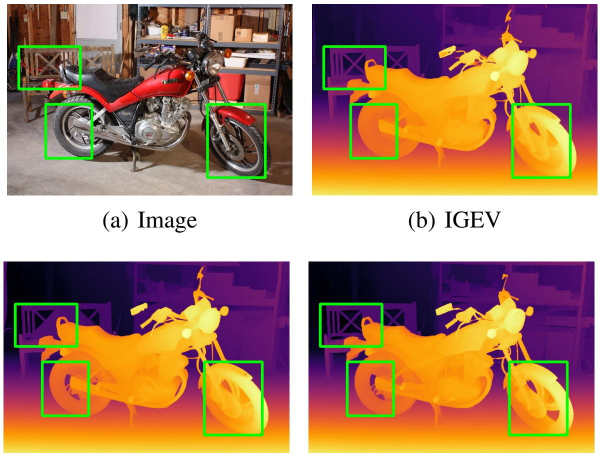

(a) Image (b) IGEV (c) DLNR (d) Ours

*Figure 1.* **Qualitative Comparison.** Comparison with state-of-the-art methods such as IGEV (Xu et al., 2023), DLNR (Zhao et al., 2023) on Middlebury dataset. All methods are trained only on SceneFlow (Mayer et al., 2016), and evaluated at quarter resolution. It can be observed that our method generalizes and predicts high-frequency details better than state-of-the-art methods.

## 1. Introduction

Stereo matching is one of the most important problems in the field of computer vision (Hoff & Ahuja, 1989; Kang et al., 1995; Scharstein & Szeliski, 2002; Szeliski, 2022). It involves the analysis of a pair of rectified stereo images, captured concurrently, with the objective of determining the pixel-level displacement map from the left image to the corresponding location in the right image, a representation commonly referred to as a "disparity map". In the context of rectified image pairs, the stereo matching problem can be conceptualized as a well-structured one-dimensional search problem in the image space (Szeliski, 2022). The utility of stereo camera systems is underscored by their efficacy and cost-effectiveness, leading to their widespread adoption in diverse commercial and industrial applications. Notably, these applications encompass domains such as autonomous navigation (Fan et al., 2020; Bimbraw, 2015), smartphone technology (Meuleman et al., 2022; Luo et al., 2020; Pang et al., 2018), and various forms of robotic vision based automation systems (Kim et al., 2021; Liu et al., 2023b;a; Jain et al., 2023; 2024; Kaya et al., 2023).

Classical stereo matching methods—often categorized as local methods, use a predefined support window to find suitable matches between stereo image pair (Scharstein & Szeliski, 2002; Hirschmuller, 2007). Yet, approaches that

[*] Equal contribution [1] Nanjing University, China. [2] ETH Zürich, Switzerland. [3] VCCM, Texas A&M University, USA. [4] UESTC, China. [5] University of Würzburg, Germany. [6] KU Leuven, Belgium. [7] INSAIT, Bulgaria. Correspondence to: Yao Yao <yaoyao@nju.edu.cn>.

*Proceedings of the $41^{st}$ International Conference on Machine Learning*, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

optimize for all disparity values using a global cost function were observed to provide better results (Kolmogorov & Zabih, 2001; Klaus et al., 2006; Bleyer et al., 2011; Yamaguchi et al., 2014). In recent years, the proliferation of high-quality, large scale synthetic ground-truth datasets, the availability of high-performance GPUs, and advancements in deep learning architectures have paved the way for deep-learning based stereo matching models trained within supervised settings. These models have shown a substantial improvement in accuracy compared to classical methods (Kendall et al., 2017a; Chang & Chen, 2018; Zhang et al., 2019; Lipson et al., 2021). Nevertheless, one fundamental challenge still remains, i.e., how to model *continuous* scene disparity values given only a limited number of candidate pixels to match? After all, the scene is continuous in nature.

Numerous recent studies have undertaken the challenge of predicting continuous scene disparities, classifiable into two main categories: ***(i)*** **Regression-based approaches** predict a real-valued offset by neural networks for each hypothesis of discrete disparity. The offset is then added to the discrete disparity hypothesis as the final continuous prediction. Typical examples include RAFT-Stereo (Lipson et al., 2021), CDN (Garg et al., 2020), and more recently IGEV (Xu et al., 2023) and DLNR (Zhao et al., 2023). ***(ii)*** **Classification-based approaches** first estimate the categorical distribution[1] for the discrete disparity hypotheses and then take the expectation value of the distribution as the final disparity, which can be any arbitrary real value even though the categorical distribution is discrete (Kendall et al., 2017a; Chang & Chen, 2018; Zhang et al., 2019).

In this paper, we aim to address the importance of continuous disparity modeling in stereo matching, given the categorical distribution of disparity hypotheses. We introduce a new perspective on the disparity prediction problem by framing it as a search problem of finding the minimum risk (Lehmann & Casella, 1998; Vapnik, 1991; Berger, 2013) of disparity values. Specifically, we define stereo risk as the average prediction error concerning all possible values of the ground-truth disparity. Since the ground-truth disparity is unavailable when making the prediction, we approximate it using the disparity hypotheses with a categorical distribution. We search for a disparity value as our prediction that achieves minimal overall risk involved with it. Furthermore, we show that the commonly used disparity expectation can be viewed as a specific instance of the $L^2$ error function of the proposed risk formulation framework. Yet, $L^2$ error function approach, despite easy to optimize, is sensitive to multi-modal distribution and leads to overly smooth solutions (Chen et al., 2019; Tosi et al., 2021). Thus, we introduce $L^1$ error function approach to risk minimization, offering potential benefit over $L^2$ limitations.

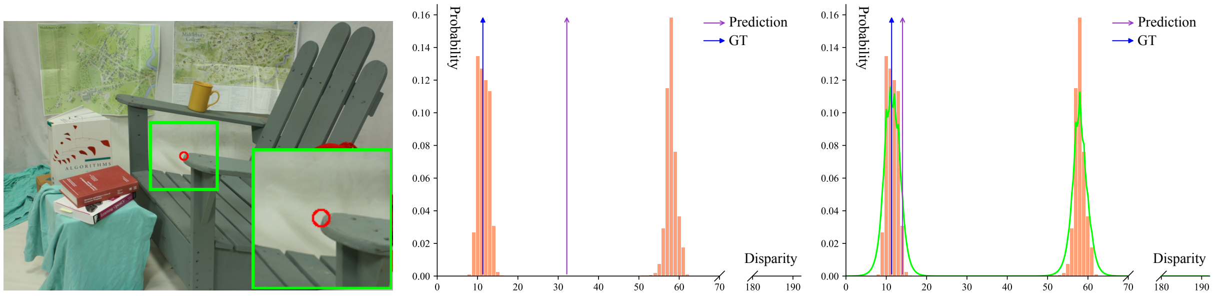
Nevertheless, our choice to use $L^1$ risk for model training leads to one practical problem, i.e., it's closed-form solution remains elusive. As a result, we embark on the pursuit of a solution by means of derivative computations applied to our novel risk function, followed by its continuous optimization. Our approach involves interpolating the disparity categorical distribution leading to defining a continuous probability density function. Subsequently, we introduce a binary search algorithm designed to efficiently identify the optimal disparity that minimizes the proposed risk. To facilitate end-to-end network training, we introduce the use the implicit function theorem (Krantz & Parks, 2002) to compute the backward gradient of the final disparity concerning the categorical distribution. All these methodological choice ensures the better model training while optimizing the proposed risk.

Upon evaluations, our approach shows superior performance compared to many state-of-the-art methods on benchmark datasets such as SceneFlow (Mayer et al., 2016), KITTI 2012 & 2015 (Geiger et al., 2012; Menze & Geiger, 2015). Moreover, our approach achieves significantly better cross-domain generalization, as observed on Middlebury (Scharstein & Szeliski, 2002), ETH 3D (Schöps et al., 2017), and KITTI 2012 & 2015. An example of qualitative comparison is given in Fig. 1. Ablation studies confirm the effectiveness of risk minimization, not only within the proposed network but also in the context of general stereo-matching networks, such as ACVNet (Xu et al., 2022) and PCWNet (Shen et al., 2022). We believe our work not only advances stereo matching in computer vision but also holds promise for its integration to robotics and control via risk analysis.

## 2. Related Work

***(i)*** **Deep Stereo Matching.** In recent years, there has been a substantial enhancement in the accuracy of stereo matching due to the adoption of deep learning-based methodologies. As a result, the pursuit of designing robust and efficient network architectures for stereo matching has emerged as a prominent area of research. For instance, Zbontar et al. (Zbontar & LeCun, 2015) harnessed deep convolutional networks to acquire discriminative features for image patches. DispNetCorr (Mayer et al., 2016) introduced explicit correlation within networks to construct cost volumes. GCNet (Kendall et al., 2017a) employed volume concatenation and refined it through 3D convolution. PSM-Net (Chang & Chen, 2018) leveraged spatial pyramid pooling (Zhao et al., 2017) and stacked hourglass networks (Newell et al., 2016) to capture contextual information. STTR (Li et al., 2021) extended the flexibility of disparity range by employing transformers (Vaswani et al., 2017; Dosovitskiy et al., 2021). Furthermore, considerations pertaining to the uniqueness

[1] A categorical distribution is a discrete probability distribution that describes the possible results of a random variable that can take on the K possible categories, with the probability of each category separately specified.



(a) Image (b) Expectation (c) Ours

*Figure 2.* **Difference between the expectation based approach and our method.** In (a) the pixel in the red circle is located at the boundary of the chair, thus the disparity distribution has multiple modes. (b) and (c) shows the discrete distribution of disparity hypotheses in orange bars. In (b) the prediction obtained by averaging is blurred and far from any of the modes. In (c) we obtained the optimal solution under $L^1$ norm, which is more robust and closer to the ground truth. The green curve is the interpolated probability density.

constraint were addressed using optimal transport (Cuturi, 2013). ACVNet (Xu et al., 2022) incorporated attention mechanisms to weight matching costs, further contributing to the advancement of stereo matching methodologies.

Another line of research is to improve efficiency. In GANet (Zhang et al., 2019) the computationally costly 3D convolutions are replaced by the differentiable semi-global aggregation (Hirschmuller, 2007). GWCNet (Guo et al., 2019) constructs the cost volume by group-wise correlation. AANet (Xu & Zhang, 2020) proposes the adaptive cost aggregation to replace the 3D convolution for efficiency. AnyNet (Wang et al., 2019), DeepPruner (Duggal et al., 2019), HITNet (Tankovich et al., 2021), CasMVSNet (Gu et al., 2020), PCWNet (Shen et al., 2022) and Bi3D (Badki et al., 2020) prune the range of disparity in the iterative manner. RAFT-Stereo (Lipson et al., 2021), CREStereo (Li et al., 2022), IGEV (Xu et al., 2023) and DLNR (Zhao et al., 2023) use recurrent neural networks (Cho et al., 2014) to predict and refine the disparity iteratively.

Inspired by CasMVSNet (Gu et al., 2020), our network consists of two stages one to predict and other to refine the disparity map. This hierarchical design reduces the time and memory cost, while keeping the matching accuracy.

***(ii)*** **Continuous Disparity by Classification.** In deep networks featuring cost volumes, the prevalent method for predicting disparity from these volumes involves the weighted average operation, commonly referred to as expectation. For instance, (Chen et al., 2019) find the average operation suffers from the over-smoothing problem, introducing the concept of a single-modal weighted average. (Garg et al., 2020) propose to predict a continuous offset to shift the distribution modes of disparity. Furthermore, they generate multi-modal ground truth disparity distributions and supervise the network to learn the distribution by Wasserstein distance. SMD-Net (Tosi et al., 2021) exploit bimodal mixture densities as output representation for disparities. UniMVSNet (Peng et al., 2022) aimed to unify the benefit of both classification and regression by introducing a novel representation and a unified focal loss. Yang et al. (Yang et al., 2022) tackled the multi-modal issue by utilizing the top-K hypotheses for disparity. On the contrary, we propose to minimize the risk under $L^1$ norm to capture continuous disparity and solve the multi-modal problem. Moreover, our approach can be trained in an end-to-end manner.

***(iii)*** **Robustness and Cross-Domain Generalization.** Existing real-world stereo datasets are small and insufficient to train deep neural networks for possible variations at test time, thereby making network robust and apt for cross-domain generalization. In this spirit, Tonioni et al. (2017; 2019a;b) fine tune the stereo matching networks on the target domain via unsupervised loss. Liu et al. (2020) jointly optimize networks for domain translation and stereo matching during training. Zhang et al. (2020); Song et al. (2021) normalize features to reduce domain shifts. Cai et al. (2020); Liu et al. (2022a) design robust features for stereo matching. Liu et al. (2022b) shows that the cost volume built by cosine similarity generalizes better. Zhang et al. (2022) apply the stereo contrastive loss and selective whitening loss to improve feature consistency. Chang et al. (2023) proposed the hierarchical visual transformation to learn invariant robust representation from synthetic images. Our approach can be combined with above methods to further improve the robustness. Yet, we present a novel perspective to improve robustness by $L^1$ risk minimization.

## 3. Method

### 3.1. Probability Modeling for Continuous Disparity

For each pixel in the left image, suppose the possible disparities are in the range of $[d_{\mathtt{min}}, d_{\mathtt{max}}]$. Conventional stereo matching algorithms typically calculate a cost function that can equate to a probability mass function (PMF) with a finite set of disparities $\mathbf{d} = [d_1, ..., d_N]^T$. It computes a discrete distribution $\mathbf{p}^m = [p_1^m, ..., p_N^m]^T$, where $d_i \in [d_{\texttt{min}}, d_{\texttt{max}}]$ and $p_i^m$ is the probability that the ground truth disparity is $d_i$. The $\mathbf{p}^m$ must satisfy $p_i^m \geq 0$ and $\sum_i p_i^m = 1$.

The discrete formulation reasons the probability only at a finite set of disparities. Yet, in real-world applications, the ground-truth disparity is continuous. Thus, we propose to interpolate the discrete distribution via Laplacian kernel, and compute the probability density function of disparity $x \in \mathbb{R}$ as

$$p(x; \mathbf{p}^m) = \sum_i^N k(x, d_i) p_i^m, \tag{1}$$

here $k(x, d_i)$ is defined as $\frac{1}{2\sigma} \exp - \frac{|x-d_i|}{\sigma}$, and $\sigma$ is the hyper-parameter for bandwidth. The above density function is valid as $p(x; \mathbf{p}^m) \geq 0$ for $\forall\, x \in \mathbb{R}$ and $\int p(x; \mathbf{p}^m) dx = 1$. An illustration of the interpolation is shown in Fig. 2 (c). The orange bars represent the given discrete distribution $\mathbf{p}^m$, and the green curve is the interpolated density function. Later, we show that such a continuous modeling enable us to compute derivative of the proposed stereo risk function.

### 3.2. Risk in Stereo Matching

To choose a value as the final prediction, we propose to minimize the following risk:

$$\texttt{argmin}_y F(y, \mathbf{p}^m) = \texttt{argmin}_y \int \mathcal{L}(y, x) p(x; \mathbf{p}^m) dx \tag{2}$$

where $F(y, \mathbf{p}^m)$ is called as the risk at $y$, and $\mathcal{L}(y, x)$ is the error function between $y$ and $x$. By risk we mean that if we take $y$ as predicted disparity, how much error there shall be with respect to the ground truth. Since the exact ground truth is unavailable at the time for making the prediction, we average the error across all possible ground-truth disparities with the distribution $p(x; \mathbf{p}^m)$.

Previous methods usually compute the expectation value of $x$ as the final prediction for the disparity:

$$y = \int x p(x; \mathbf{p}^m) dx. \tag{3}$$

We want to point out that we can arrive at the same prediction by using squared $L^2$ norm loss as $\mathcal{L}(y, x)$ in Eq.(2), i.e., $\texttt{argmin}_y F(y, \mathbf{p}^m) = \int x p(x; \mathbf{p}^m) dx$ if $\mathcal{L}(y, x) = (y - x)^2$. Nevertheless, it is well known that the $L^2$ norm is not robust to outliers. As an example, in Fig. 2 (b) it can be observed that the estimated expectation is inaccurate when there are multiple modes in the distribution. And therefore, we resort to $L^1$ norm of $\mathcal{L}(y, x)$ in Eq.(2), i.e.,

$$\texttt{argmin}_y F(y, \mathbf{p}^m) = \texttt{argmin}_y \int |y - x| p(x; \mathbf{p}^m) dx. \tag{4}$$

**Algorithm 1** Forward Prediction

```
input $\tau > 0$, $\sigma > 0$, $\mathbf{d} = [d_1, ..., d_N]$, $d_1 < d_2 < \cdots < d_N$, and $\mathbf{p}^m = [p_1^m, ..., p_N^m]$
  $d^l \leftarrow d_1$
  $d^r \leftarrow d_N$
  $g \leftarrow \tau + 1$
  while $|g| > \tau$ do
    $d^m \leftarrow (d^l + d^r)/2.0$
    $g \leftarrow \sum_i p_i^m \texttt{Sign}(d^m - d_i)(1 - \exp - \frac{|d^m - d_i|}{\sigma})$
    if $g > 0$ then
      $d^r \leftarrow d^m$
    else
      $d^l \leftarrow d^m$
    end if
  end while
output $d^m$
```

Given the distribution $p(x; \mathbf{p}^m)$ of the disparity, the optimal $y$ will minimize the $L^1$ error with respect to all possible disparities weighted by the corresponding probability density. As shown in Fig. 2 (c), our final prediction is more robust to the incorrect modes and closer to the ground truth.

### 3.3. Differentiable Stereo Risk Minimization

Obtaining a minimal risk solution to Eq.(4) seems difficult as it is challenging to derive its closed form formulation. So, performing end-to-end learning with deep network seems difficult. To this end, we put forward an approach that enable end-to-end learning of the network as follows:

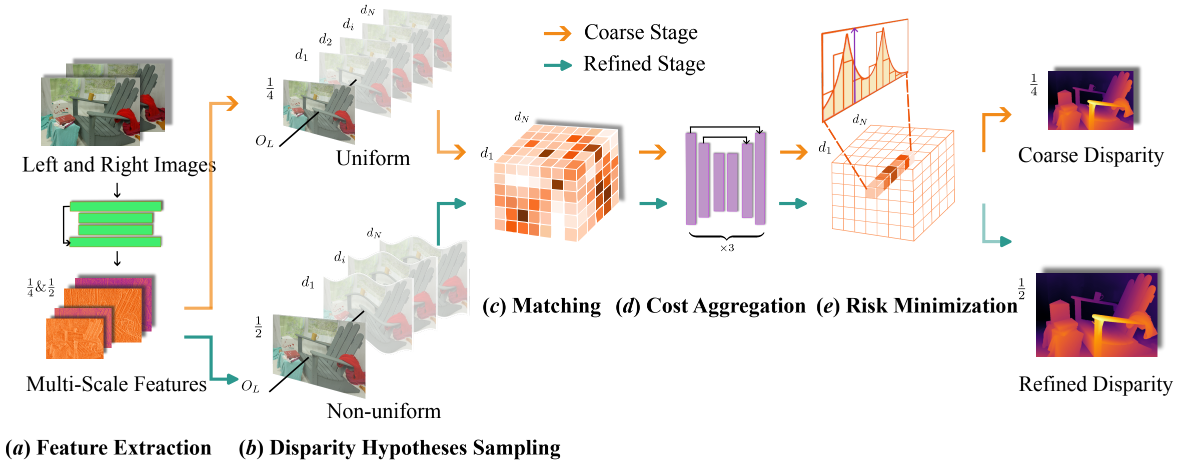
***(i)* Forward Prediction.** Given a discrete distribution $\mathbf{p}^m$, we find the optimal $y$ for Eq.(4) based on the following two observations. Firstly, the target function $F(y, \mathbf{p}^m)$ is convex with respect to $y$, hence we compute the optimal solution where $\partial F / \partial y = 0$, i.e.,

$$G(y, \mathbf{p}^m) \triangleq \frac{\partial F(y, \mathbf{p}^m)}{\partial y} \tag{5}$$

$$= \sum_i p_i^m \texttt{Sign}(y - d_i)(1 - \exp - \frac{|y - d_i|}{\sigma}) = 0 \tag{6}$$

where $\texttt{Sign}()$ is the sign function (a slight abuse of notation). $\texttt{Sign}()$ can be thought of as an indicator function, i.e., it is 1 if $y > d_i$ and $-1$ otherwise. Secondly, the second-order derivative $\partial^2 F / \partial^2 y \geq 0$, indicating that the first-order derivative is a non-decreasing function. We find the optimal disparity, i.e., the zero point of $G(y, \mathbf{p}^m)$, by binary search, as shown in Algorithm 1. In all our experiments, we set $\sigma = 1.1$ and $\tau = 0.1$. For $N$ disparity hypotheses, the binary search algorithm can find the optimal solution with time complexity of $O(\log N)$ (Cormen et al., 2009).

***(ii)* Backward Propagation.** Our approach to forward prediction for solving Eq.(4) contains non-differentiable op-



**(*a*) Feature Extraction** **(*b*) Disparity Hypotheses Sampling**

*Figure 3.* **Overall pipeline (Left to Right).** We first extract multi-scale features from left and right images respectively. The subsequent procedures are divided into two stages. In the coarse stage —shown in orange arrow, we sample disparity hypotheses uniformly and match on 1/4-resolution features. While in the refined stage—shown in green arrow, to match 1/2-resolution features efficiently. Disparity hypotheses are sampled centering around the disparity predicted from the coarse stage. In both stages, we first construct cost volumes by concatenation, and then apply the stacked hourglass networks to aggregate the matching cost, and finally search for the disparity that minimizes the proposed $L^1$ risk in Eq.(4).

erations —refer Algorithm 1. Yet, to enable end-to-end training, we have to compute $dy/d\mathbf{p}^m$ to propagate the gradient backward. Now, since $G(y, \mathbf{p}^m) \equiv 0$ at the optimal $y$, we obtain the following via use of Implicit Function Theorem (Krantz & Parks, 2002):

$$dG(y, \mathbf{p}^m) = \frac{\partial G}{\partial y} dy + \frac{\partial G}{\partial \mathbf{p}^m} d\mathbf{p}^m = 0. \quad (7)$$

By organizing the terms, we obtain

$$\frac{dy}{d\mathbf{p}^m} = -\frac{\partial G / \partial \mathbf{p}^m}{\partial G / \partial y} \quad (8)$$

$$= [\ldots, \frac{\sigma \mathtt{Sign}(d_i - y)(1 - \exp -\frac{|y-d_i|}{\sigma})}{\sum_j p_j^m \exp -\frac{|y-d_j|}{\sigma}}, \ldots]^T, \quad (9)$$

showing the back-propagation computation. Here, we clip the denominator $\sum_j p_j^m \exp -\frac{|y-d_j|}{\sigma}$ in the above equation to be no less than 0.1 to avoid large gradients.

### 3.4. Network Architecture

To find the disparity value, we match the image patches of left and right images by constructing stereo cost volumes as done in Kendall et al. (2017b); Chang & Chen (2018). Yet, an exhaustive matching requires extensive memory and computation. So, for efficiency, we adopt a cascade structure following Gu et al. (2020). Categorically, we first sample the disparity hypothesis by a coarse matching at low-resolution image features. This reduces the search space to a large extent. Next, we refine the sampled hypothesis at high-resolution image features.

Fig. 3 shows the overall network architecture details. For clarity on our design choices, we explain the network components in five module as follows[2]:

***(a)* Feature Extraction.** Given an input image, the module aims to output multi-scale 2D feature maps. Specifically, we first use a ResNet (He et al., 2016) to extract 2D feature maps of resolution 1/4 and 1/2 with respect to the input image. The ResNet contains 4 stages of transformation with 3, 16, 3, 3 residual blocks, respectively. The spatial resolution is downsampled before the beginning of the first and third stages of transformation. Next, we apply the spatial pyramid pooling (Zhao et al., 2017) on the 1/4-resolution feature map from the fourth stage to enlarge the receptive field. In the end, we upsample the enhanced feature map from 1/4 to 1/2 and fuse it with the 1/2-resolution feature map from ResNet. The final outputs are the feature maps of 1/4 and 1/2 resolution. We apply the same network and weights to extract features from left and right images.

***(b)* Disparity Hypotheses Sampling.** The disparity hypotheses provide the candidates of pixel pairs to match. In the coarse stage, we uniformly sample 192 hypotheses in the range 0 to maximum possible disparity. In the refined stage, we reduce the sampling space according to the predicted disparity from the coarse stage. Concretely, for each pixel we sample 16 hypotheses between the minimum and maximum disparity in the local window of size 12 × 12.

***(c)* Matching.** We match the 2D feature maps from the left and right images according to the sampled disparity

[2] More details are provided in the Appendix

*Table 1.* Comparison with state-of-the-art on SceneFlow test set. The $1^{st}$ and $2^{nd}$ bests are in red and blue, respectively. **Ours** in bold.

| Method | Param (M) | Time (s) | EPE ↓ | > 0.5px ↓ | > 1px ↓ | > 2px ↓ |
|---|---|---|---|---|---|---|
| CFNet (Shen et al., 2021) | 21.98 | 0.13 | 1.04 | 15.91 | 10.30 | 6.89 |
| PCWNet (Shen et al., 2022) | 34.27 | 0.25 | 0.90 | 17.59 | 8.08 | 4.57 |
| ACVNet (Xu et al., 2022) | 6.84 | 0.16 | 0.47 | 9.70 | 5.00 | 2.74 |
| DLNR (Zhao et al., 2023) | 54.72 | 0.44 | 0.53 | 8.75 | 5.44 | 3.44 |
| IGEV (Xu et al., 2023) | 12.60 | 0.36 | 0.47 | 8.51 | 5.21 | 3.26 |
| **Ours** | **11.96** | **0.35** | **0.43** | **8.10** | **4.22** | **2.34** |

*Table 2.* Comparison with state-of-the-art methods on KITTI 2012 Benchmark. † denotes using extra data for pre-training. The first and second bests are in red and blue respectively. **Our method** in bold. The results are obtained from KITTI official website.

| Method | Param (M) | Time (s) | > 2px | | > 3px | |
|---|---|---|---|---|---|---|
| | | | Noc | All | Noc | All |
| LEAStereo (Cheng et al., 2020) | 1.81 | | 1.90 | 2.39 | 1.13 | 1.45 |
| CFNet (Shen et al., 2021) | 21.98 | 0.12 | 1.90 | 2.43 | 1.23 | 1.58 |
| ACVNet (Xu et al., 2022) | 6.84 | 0.15 | 1.83 | 2.34 | 1.13 | 1.47 |
| ACFNet (Chen et al., 2021) | | | 1.83 | 2.35 | 1.17 | 1.54 |
| NLCA-Net v2 (Rao et al., 2022) | | | 1.83 | 2.34 | 1.11 | 1.46 |
| CAL-Net (Chen et al., 2021) | | | 1.74 | 2.24 | 1.19 | 1.53 |
| CREStereo (Li et al., 2022) † | | | 1.72 | 2.18 | 1.14 | 1.46 |
| LaC+GANet (Liu et al., 2022a) | 9.43 | | 1.72 | 2.26 | 1.05 | 1.42 |
| IGEV (Xu et al., 2023)† | 12.60 | 0.32 | 1.71 | 2.17 | 1.12 | 1.44 |
| PCWNet (Shen et al., 2022) | 34.27 | 0.23 | 1.69 | 2.18 | 1.04 | 1.37 |
| **Ours** | **11.96** | **0.32** | **1.58** | **2.20** | **1.00** | **1.44** |

hypothesis. The features at each pair of candidates pixels for matching will be concatenated along the channel dimension, which forms a 4D stereo cost volume (feature× disparity×height×width). In the coarse stage, we match the feature map of 1/4 resolution for efficiency. To capture high-frequency details, we match the 1/2-resolution feature map in the refined stage.

***(d)* Cost Aggregation.** We use the stacked hourglass architecture (Newell et al., 2016) to transform the stereo cost volume and aggregate the matching cost. For the coarse and refined stages, the structures are same except for the number of feature channels. Specifically, the network consists of three 3D hourglasss as in (Chang & Chen, 2018). Each hourglass first downsamples the volume hierarchically to 1/2 and 1/4 resolution with respect to the input volume, and then upsample in sequence to recover the resolution. This procedure helps aggregate information across various scales. The final output is a volume that represents the discrete distribution of disparity hypotheses.

***(e)* Risk Minimization.** This module applies Alg. 1 to compute the optimal continuous disparity for each pixel given the discrete distribution of disparity hypotheses. At train time, we additionally compute the gradient according to Eq.(9) to enable backward propagation.

### 3.5. Loss Function

Given the predicted disparity $x^{\text{pred}} \in \mathbb{R}$ and the ground-truth disparity $x^{\text{gt}} \in \mathbb{R}$, we compute the smooth $L^1$ loss as

$$\mathcal{L}(x^{\text{gt}}, x^{\text{pred}}) = \begin{cases} 0.5(x^{\text{gt}} - x^{\text{pred}})^2, & \text{if } |x^{\text{gt}} - x^{\text{pred}}| < 1.0 \\ |x^{\text{gt}} - x^{\text{pred}}| - 0.5, & \text{otherwise} \end{cases} \tag{10}$$

We apply the above loss function to the predicted disparities from both the coarse and refined stages, and obtain $\mathcal{L}_{\texttt{coarse}}$ and $\mathcal{L}_{\texttt{refined}}$, respectively. The total loss is thus defined as $\mathcal{L} = 0.1 * \mathcal{L}_{\texttt{coarse}} + 1.0 * \mathcal{L}_{\texttt{refined}}$.

## 4. Experiments and Results

**Implementation Details.** We implement our method in PyTorch 2.0.1 (Python 3.11.2) with CUDA 11.8. The software is evaluated on a machine with GeForce-RTX-3090 GPU.

**Datasets.** We perform experiments on four datasets namely SceneFlow (Mayer et al., 2016), KITTI 2012 & 2015 (Geiger et al., 2012; Menze & Geiger, 2015), Middlebury 2014 (Scharstein & Szeliski, 2002), and ETH 3D (Schöps et al., 2017). **(a) SceneFlow** is a synthetic dataset containing 35,454 image pairs for training, and 4,370 image pairs for test. **(b) KITTI 2012 & 2015** are captured for autonomous driving. There are 194 training image pairs and 195 test image pairs in KITTI 2012. For KITTI 2015, there are 200

*Table 3.* Comparison with state-of-the-art methods on KITTI 2015 Benchmark. † denotes using extra data for pre-training. The first and second bests are in red and blue respectively. **Our method** in bold. The results are obtained from KITTI official website.

| Method | Param (M) | Time (s) | All | | | Noc | | |
|---|---|---|---|---|---|---|---|---|
| | | | D1_bg | D1_fg | D1_all | D1_bg | D1_fg | D1_all |
| LEAStereo (Cheng et al., 2020) | 1.81 | | 1.40 | 2.91 | 1.65 | 1.29 | 2.65 | 1.51 |
| CFNet (Shen et al., 2021) | 21.98 | 0.12 | 1.54 | 3.56 | 1.88 | 1.43 | 3.25 | 1.73 |
| ACVNet (Xu et al., 2022) | 6.84 | 0.15 | 1.37 | 3.07 | 1.65 | 1.26 | 2.84 | 1.52 |
| ACFNet (Chen et al., 2021) | | | 1.51 | 3.80 | 1.89 | 1.36 | 3.49 | 1.72 |
| NLCA-Net v2 (Rao et al., 2022) | | | 1.41 | 3.56 | 1.77 | 1.28 | 3.22 | 1.60 |
| CAL-Net (Chen et al., 2021) | | | 1.59 | 3.76 | 1.95 | 1.45 | 3.42 | 1.77 |
| CREStereo (Li et al., 2022) † | | | 1.45 | 2.86 | 1.69 | 1.33 | 2.60 | 1.54 |
| LaC+GANet (Liu et al., 2022a) | 9.43 | | 1.44 | 2.83 | 1.67 | 1.26 | 2.64 | 1.49 |
| IGEV (Xu et al., 2023) † | 12.60 | 0.32 | 1.38 | 2.67 | 1.59 | 1.27 | 2.62 | 1.49 |
| DLNR (Zhao et al., 2023) | 54.72 | 0.39 | 1.60 | 2.59 | 1.76 | 1.45 | 2.39 | 1.61 |
| PCWNet (Shen et al., 2022) | 34.27 | 0.23 | 1.37 | 3.16 | 1.67 | 1.26 | 2.93 | 1.53 |
| CroCo (Weinzaepfel et al., 2023)† | 417.15 | | 1.38 | 2.65 | 1.59 | 1.30 | 2.56 | 1.51 |
| **Ours** | **11.96** | **0.32** | **1.40** | **2.76** | **1.63** | **1.25** | **2.62** | **1.48** |

training image pairs and 200 test image pairs. **(c) Middlebury 2014** is an indoor dataset including 15 image pairs for training. **(d) ETH 3D** is a gray-scale dataset providing 27 image pairs for training.

**Training Details.** We train our network on SceneFlow. The weight is initialized randomly. We use AdamW optimizer (Loshchilov & Hutter, 2019) with weight decay $10^{-5}$. The learning rate decreases from $2 \times 10^{-4}$ to $2 \times 10^{-8}$ according to the one cycle learning rate policy. We train the network for $2 \times 10^5$ iterations. The images are randomly cropped to $320 \times 736$. For KITTI 2012 & 2015 benchmarks, we further fine tune the network on the training image pairs for $2.5 \times 10^3$ iterations. The learning rate goes from $5 \times 10^{-5}$ to $5 \times 10^{-9}$ over iterations. More details are provided in the Appendix.

*Table 4.* Cross-domain evaluation on Middlebury train set of quarter resolution. † denotes using extra data for pre-training. The first and second bests are in red and blue respectively. All methods are trained on SceneFlow and evaluated on Middlebury train set without fine-tuning.

| Method | > 0.5px | | > 1px | |
|---|---|---|---|---|
| | Noc | All | Noc | All |
| CFNet | 29.50 | 34.30 | 17.85 | 22.16 |
| ACVNet | 39.04 | 42.97 | 22.68 | 26.49 |
| DLNR | 19.43 | 23.75 | 10.16 | 13.76 |
| IGEV † | 19.05 | 23.33 | 10.44 | 14.05 |
| PCWNet | 33.33 | 38.00 | 16.80 | 21.36 |
| **Ours** | **19.22** | **23.33** | **9.32** | **12.63** |

### 4.1. In-Domain Evaluation

Tab.(1), Tab.(2) and Tab.(3) provide statistical comparison results with the competing methods on SceneFlow, KITTI 2012, and KITTI 2015 bechmarks, respectively. All the methods have been trained and fine-tuned on the corresponding training set. For SceneFlow test set, our proposed approach shows the best results over all the evaluation metrics. Particularly, we reduce $>$ 1px error from 5.00 to 4.22, and $>$ 0.5px error from 8.51 to 8.10. For KITTI 2012 & 2015 benchmarks, the matching accuracy of our approach in the non-occluded regions rank the first among the published methods. Especially, in KITTI 2012, we reduce the $>$ 2px error in non-occluded regions by 0.11.

*Table 5.* Cross-domain evaluation on ETH 3D train set. † denotes using extra data for pre-training. The first and second bests are in red and blue respectively. All methods are trained on SceneFlow and evaluated on ETH 3D train set without fine-tuning.

| Method | > 0.5px | | > 1px | |
|---|---|---|---|---|
| | Noc | All | Noc | All |
| CFNet | 15.57 | 16.24 | 5.30 | 5.59 |
| ACVNet | 21.83 | 22.64 | 8.13 | 8.81 |
| DLNR | 18.66 | 19.07 | 13.11 | 13.39 |
| IGEV † | 9.83 | 10.39 | 3.60 | 4.05 |
| PCWNet | 18.25 | 18.88 | 5.17 | 5.43 |
| **Ours** | **7.90** | **8.59** | **2.41** | **2.71** |

### 4.2. Cross-Domain Generalization

For this experiment, we compare the methods when dealing with environments never seen in the train set. Specifically, all methods are trained only on SceneFlow training set, and then evaluated on the Middlebury, ETH 3D and KITTI 2012 & 2015 train set, "without" fine-tuning.

The statistical comparison results are shown in Tab.(4), Tab.(5), Tab.(7), and Tab.(8). Our proposed approach achieves the first or the second best accuracies under all the evaluation metrics on the 4 real-world datasets. Particu-

*Table 6.* Ablation studies on Middlebury training set of quarter resolution. The first and second bests are in red and blue respectively. **Our method** in bold. All methods are trained on SceneFlow and evaluated on Middlebury training set without fine-tuning.

| Backbone | Training | Test | Param (M) | Time (s) | $> 1$px |  | $> 2$px |  |
|---|---|---|---|---|---|---|---|---|
|  |  |  |  |  | Noc | All | Noc | All |
| ACVNet | Expectation | Expectation | 6.84 | 0.12 | 22.68 | 26.49 | 13.54 | 16.49 |
|  | Expectation | L1-Risk | 6.84 | 0.18 | 22.32 | 26.14 | 13.13 | 16.05 |
| PCWNet | Expectation | Expectation | 34.27 | 0.19 | 16.80 | 21.36 | 8.93 | 12.62 |
|  | Expectation | L1-Risk | 34.27 | 0.26 | 16.53 | 21.08 | 8.65 | 12.30 |
| **Ours** | Expectation | Expectation | 11.96 | 0.17 | 9.88 | 13.27 | 4.92 | 7.29 |
|  | Expectation | L1-Risk | 11.96 | 0.25 | 9.83 | 13.22 | 4.90 | 7.27 |
|  | L1-Risk | Expectation | 11.96 | 0.17 | 9.83 | 13.19 | 4.79 | 7.06 |
|  | **L1-Risk** | **L1-Risk** | **11.96** | **0.25** | **9.32** | **12.63** | **4.49** | **6.70** |

*Table 7.* Cross-domain evaluation on KITTI 2012 train set. † denotes using extra data for pre-training. The first and second bests are in red and blue respectively. All methods are trained on SceneFlow and evaluated on KITTI 2012 train set without fine-tuning.

| Method | $> 2$px |  | $> 3$px |  |
|---|---|---|---|---|
|  | Noc | All | Noc | All |
| CFNet | 7.08 | 7.97 | 4.66 | 5.31 |
| ACVNet | 20.34 | 21.44 | 14.22 | 15.18 |
| DLNR | 12.01 | 12.81 | 8.83 | 9.46 |
| IGEV | 7.55 | 8.44 | 5.03 | 5.70 |
| PCWNet | 6.63 | 7.49 | 4.08 | 4.68 |
| **Ours** | **5.82** | **6.70** | **3.84** | **4.43** |

*Table 8.* Cross-domain evaluation on KITTI 2015 train set. † denotes using extra data for pre-training. The first and second bests are in red and blue respectively. All methods are trained on SceneFlow and evaluated on KITTI 2015 train set without fine-tuning.

| Method | All |  |  |
|---|---|---|---|
|  | D1_bg | D1_fg | D1_all |
| CFNet | 4.77 | 13.26 | 6.07 |
| ACVNet | 12.35 | 19.97 | 13.52 |
| DLNR | 18.67 | 14.86 | 18.08 |
| IGEV | 4.01 | 15.58 | 5.79 |
| PCWNet | 4.25 | 14.40 | 5.81 |
| **Ours** | **3.68** | **13.52** | **5.19** |

larly, for Middlebury, we reduce the $>$ 1px error from 13.76 to 12.63. Furthermore, on ETH 3D we reduce $>$ 0.5px error from 10.39 to 8.59, and $>$ 1px error from 4.05 to 2.71. Thus, our approach result seems more resilient to cross-domain setting and generalizes better than competing methods. The qualitative comparison is provided in the Appendix.

### 4.3. Ablation Studies

We performed ablations to analyze risk minimization effects in disparity prediction. All the models are trained on SceneFlow and tested on Middlebury without fine-tuning.

***(a)* Effect of Risk Minimization.** We compare the expectation, i.e., Eq.(2) and the $L^1$-norm risk minimization for disparity prediction at train and test time. We present the comparison results in Tab.(6). Even with expectation minimization at train time, we slightly improve the matching accuracy with $L^1$-norm risk minimization at test time. Yet, if we use the $L^1$-norm risk minimization at both train time and test time, the best accuracy is achieved under all metrics.

***(b)* Performance with Different Networks.** We replace the disparity prediction method in ACVNet (Xu et al., 2022) and PCWNet (Shen et al., 2022) from expectation i.e., Eq.(2) to $L^1$-norm risk minimization *only* during test. The results are shown in Tab.(6). Our proposed method improves the accuracy under all metrics *without* re-training.

### 4.4. Network Processing Time & Paremeters

We present the networks' inference time and number of parameters in Tab.(1), Tab.(2), Tab.(3), and Tab.(6)—cf. Time (s) column. For a fair comparison, all networks are evalutated on the same machine with a GeForce-RTX-3090 GPU. Our network outperforms many state of the arts on inference time, including IGEV and DLNR. Moreover, our network has fewer learnable parameters than PCWNet, IGEV and DLNR. In addition, our proposed $L^1$-norm risk minimization module doesn't require extra learnable parameters. The running time is shown in Tab.(6). By changing the disparity prediction method from expectation minimization to our proposed approach, the running time increases slightly.

## 5. Conclusion

The paper concludes that continuous end-to-end trainable model for stereo matching is possible with $L^1$ risk minimization formulation. It is shown that the proposed approach is beneficial to multi-modal disparity distributions and outliers and generalizes better on cross-domain stereo images. Stereo Risk is unique in a way that it provides a new way of solving stereo-matching with well-thought-out theoretical arc (Vapnik, 1991) and improved results, enabling adaptations from fields such as robotics and control engineering.

## Impact Statement

We include the following statement, in accordance with the ICML 2024 guidelines outlined at https://icml.cc/Conferences/2024/CallForPapers in the Impact Statement section.

**Stereo matching**, a key technology in computer vision, promises significant advancements in areas like autonomous vehicles, medical imaging, virtual reality, and robotics, enhancing safety, efficiency, and immersive experiences. The future of stereo matching will significantly impact society, specifically coming from the automation industry and therefore, it requires a balanced approach in its usage that maximizes benefits while mitigating risks, ensuring its development aligns with societal values and needs.

## Acknowledgements

This work was partly supported by the Alexander von Humboldt Foundation and NSFC 62441204.

## References

Badki, A., Troccoli, A., Kim, K., Kautz, J., Sen, P., and Gallo, O. Bi3d: Stereo depth estimation via binary classifications. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1600–1608, 2020.

Berger, J. O. *Statistical decision theory and Bayesian analysis*. Springer Science & Business Media, 2013.

Bimbraw, K. Autonomous cars: Past, present and future a review of the developments in the last century, the present scenario and the expected future of autonomous vehicle technology. In *2015 12th international conference on informatics in control, automation and robotics (ICINCO)*, volume 1, pp. 191–198. IEEE, 2015.

Bleyer, M., Rhemann, C., and Rother, C. Patchmatch stereo-stereo matching with slanted support windows. In *Bmvc*, volume 11, pp. 1–11, 2011.

Cai, C., Poggi, M., Mattoccia, S., and Mordohai, P. Matching-space stereo networks for cross-domain generalization. In *2020 International Conference on 3D Vision (3DV)*, pp. 364–373. IEEE, 2020.

Chang, J.-R. and Chen, Y.-S. Pyramid stereo matching network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5410–5418, 2018.

Chang, T., Yang, X., Zhang, T., and Wang, M. Domain generalized stereo matching via hierarchical visual transformation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9559–9568, 2023.

Chen, C., Chen, X., and Cheng, H. On the over-smoothing problem of cnn based disparity estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Chen, S., Li, B., Wang, W., Zhang, H., Li, H., and Wang, Z. Cost affinity learning network for stereo matching. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2120–2124. IEEE, 2021.

Cheng, X., Zhong, Y., Harandi, M., Dai, Y., Chang, X., Li, H., Drummond, T., and Ge, Z. Hierarchical neural architecture search for deep stereo matching. *Advances in Neural Information Processing Systems*, 33:22158–22169, 2020.

Cho, K., Merrienboer, B., Gulcehre, C., Bougares, F., Schwenk, H., and Bengio, Y. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *EMNLP*, 2014.

Chuah, W., Tennakoon, R., Hoseinnezhad, R., Bab-Hadiashar, A., and Suter, D. Itsa: An information-theoretic approach to automatic shortcut avoidance and domain generalization in stereo matching networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13022–13032, June 2022.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. *Introduction to Algorithms, Third Edition*. The MIT Press, 3rd edition, 2009. ISBN 0262033844.

Cuturi, M. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Duggal, S., Wang, S., Ma, W.-C., Hu, R., and Urtasun, R. Deeppruner: Learning efficient stereo matching via differentiable patchmatch. In *ICCV*, 2019.

Fan, R., Wang, L., Bocus, M. J., and Pitas, I. Computer stereo vision for autonomous driving. *arXiv preprint arXiv:2012.03194*, 2020.

Garg, D., Wang, Y., Hariharan, B., Campbell, M., Weinberger, K. Q., and Chao, W.-L. Wasserstein distances for stereo disparity estimation. *Advances in Neural Information Processing Systems*, 33:22517–22529, 2020.

Geiger, A., Lenz, P., and Urtasun, R. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, pp. 3354–3361. IEEE, 2012.

Gu, X., Fan, Z., Zhu, S., Dai, Z., Tan, F., and Tan, P. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2495–2504, 2020.

Guo, X., Yang, K., Yang, W., Wang, X., and Li, H. Group-wise correlation stereo network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3273–3282, 2019.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Hirschmuller, H. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelligence*, 30(2):328–341, 2007.

Hoff, W. and Ahuja, N. Surfaces from stereo: Integrating feature matching, disparity estimation, and contour detection. *IEEE transactions on pattern analysis and machine intelligence*, 11(2):121–136, 1989.

Jain, N., Kumar, S., and Van Gool, L. Enhanced stable view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13208–13217, 2023.

Jain, N., Kumar, S., and Van Gool, L. Learning robust multi-scale representation for neural radiance fields from unposed images. *International Journal of Computer Vision*, 132(4):1310–1335, 2024.

Kang, S. B., Webb, J., Zitnick, C., and Kanade, T. A multi-baseline stereo system with active illumination and real-time image acquisition. In *Proceedings of IEEE International Conference on Computer Vision*, pp. 88–93, 1995. doi: 10.1109/ICCV.1995.466802.

Kaya, B., Kumar, S., Oliveira, C., Ferrari, V., and Van Gool, L. Multi-view photometric stereo revisited. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3126–3135, 2023.

Kendall, A., Martirosyan, H., Dasgupta, S., Henry, P., Kennedy, R., Bachrach, A., and Bry, A. End-to-end learning of geometry and context for deep stereo regression. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017a.

Kendall, A., Martirosyan, H., Dasgupta, S., Henry, P., Kennedy, R., Bachrach, A., and Bry, A. End-to-end learning of geometry and context for deep stereo regression. In *Proceedings of the IEEE international conference on computer vision*, pp. 66–75, 2017b.

Kim, W.-S., Lee, D.-H., Kim, Y.-J., Kim, T., Lee, W.-S., and Choi, C.-H. Stereo-vision-based crop height estimation for agricultural robots. *Computers and Electronics in Agriculture*, 181:105937, 2021.

Klaus, A., Sormann, M., and Karner, K. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 3, pp. 15–18. IEEE, 2006.

Kolmogorov, V. and Zabih, R. Computing visual correspondence with occlusions using graph cuts. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pp. 508–515. IEEE, 2001.

Krantz, S. G. and Parks, H. R. *The implicit function theorem: history, theory, and applications*. Springer Science & Business Media, 2002.

Lehmann, E. L. and Casella, G. *Theory of Point Estimation*. Springer-Verlag, New York, NY, USA, second edition, 1998.

Li, J., Wang, P., Xiong, P., Cai, T., Yan, Z., Yang, L., Liu, J., Fan, H., and Liu, S. Practical stereo matching via cascaded recurrent network with adaptive correlation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16263–16272, 2022.

Li, Z., Liu, X., Drenkow, N., Ding, A., Creighton, F. X., Taylor, R. H., and Unberath, M. Revisiting stereo depth estimation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6197–6206, October 2021.

Lipson, L., Teed, Z., and Deng, J. Raft-stereo: Multilevel recurrent field transforms for stereo matching. In *2021 International Conference on 3D Vision (3DV)*, pp. 218–227. IEEE, 2021.

Liu, B., Yu, H., and Long, Y. Local similarity pattern and cost self-reassembling for deep stereo matching networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 1647–1655, 2022a.

Liu, B., Yu, H., and Qi, G. Graftnet: Towards domain generalized stereo matching with a broad-spectrum and task-oriented feature. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13012–13021, 2022b.

Liu, C., Kumar, S., Gu, S., Timofte, R., and Van Gool, L. Single image depth prediction made better: A multivariate gaussian take. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17346–17356, 2023a.

Liu, C., Kumar, S., Gu, S., Timofte, R., and Van Gool, L. Vadepthnet: A variational approach to single image depth prediction. *arXiv preprint arXiv:2302.06556*, 2023b.

Liu, R., Yang, C., Sun, W., Wang, X., and Li, H. Stereogan: Bridging synthetic-to-real domain gap by joint optimization of domain translation and stereo matching. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Luo, C., Li, Y., Lin, K., Chen, G., Lee, S.-J., Choi, J., Yoo, Y. F., and Polley, M. O. Wavelet synthesis net for disparity estimation to synthesize dslr calibre bokeh effect on smartphones. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2407–2415, 2020.

Mayer, N., Ilg, E., Hausser, P., Fischer, P., Cremers, D., Dosovitskiy, A., and Brox, T. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4040–4048, 2016.

Menze, M. and Geiger, A. Object scene flow for autonomous vehicles. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3061–3070, 2015.

Meuleman, A., Kim, H., Tompkin, J., and Kim, M. H. Floatingfusion: Depth from tof and image-stabilized stereo cameras. In *European Conference on Computer Vision*, pp. 602–618. Springer, 2022.

Newell, A., Yang, K., and Deng, J. Stacked hourglass networks for human pose estimation. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14*, pp. 483–499. Springer, 2016.

Pang, J., Sun, W., Yang, C., Ren, J., Xiao, R., Zeng, J., and Lin, L. Zoom and learn: Generalizing deep stereo matching to novel domains. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2070–2079, 2018.

Peng, R., Wang, R., Wang, Z., Lai, Y., and Wang, R. Rethinking depth estimation for multi-view stereo: A unified representation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Rao, Z., Dai, Y., Shen, Z., and He, R. Rethinking training strategy in stereo matching. *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–14, 2022.

Scharstein, D. and Szeliski, R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International journal of computer vision*, 47:7–42, 2002.

Schöps, T., Schönberger, J. L., Galliani, S., Sattler, T., Schindler, K., Pollefeys, M., and Geiger, A. A multiview stereo benchmark with high-resolution images and multi-camera videos. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Shen, Z., Dai, Y., and Rao, Z. Cfnet: Cascade and fused cost volume for robust stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13906–13915, 2021.

Shen, Z., Dai, Y., Song, X., Rao, Z., Zhou, D., and Zhang, L. Pcw-net: Pyramid combination and warping cost volume for stereo matching. In *European Conference on Computer Vision*, pp. 280–297. Springer, 2022.

Song, X., Yang, G., Zhu, X., Zhou, H., Wang, Z., and Shi, J. Adastereo: A simple and efficient approach for adaptive stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10328–10337, June 2021.

Szeliski, R. *Computer vision: algorithms and applications*. Springer Nature, 2022.

Tankovich, V., Hane, C., Zhang, Y., Kowdle, A., Fanello, S., and Bouaziz, S. Hitnet: Hierarchical iterative tile refinement network for real-time stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14362–14372, 2021.

Tonioni, A., Poggi, M., Mattoccia, S., and Di Stefano, L. Unsupervised adaptation for deep stereo. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

Tonioni, A., Rahnama, O., Joy, T., Stefano, L. D., Ajanthan, T., and Torr, P. H. Learning to adapt for stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9661–9670, 2019a.

Tonioni, A., Tosi, F., Poggi, M., Mattoccia, S., and Stefano, L. D. Real-time self-adaptive deep stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019b.

Tosi, F., Liao, Y., Schmitt, C., and Geiger, A. Smd-nets: Stereo mixture density networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8942–8952, June 2021.

Vapnik, V. Principles of risk minimization for learning theory. *Advances in neural information processing systems*, 4, 1991.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Wang, Y., Lai, Z., Huang, G., Wang, B. H., Van Der Maaten, L., Campbell, M., and Weinberger, K. Q. Anytime stereo image depth estimation on mobile devices. In *2019 international conference on robotics and automation (ICRA)*, pp. 5893–5900. IEEE, 2019.

Weinzaepfel, P., Lucas, T., Leroy, V., Cabon, Y., Arora, V., Brégier, R., Csurka, G., Antsfeld, L., Chidlovskii, B., and Revaud, J. CroCo v2: Improved Cross-view Completion Pre-training for Stereo Matching and Optical Flow. In *ICCV*, 2023.

Xu, G., Cheng, J., Guo, P., and Yang, X. Attention concatenation volume for accurate and efficient stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12981–12990, 2022.

Xu, G., Wang, X., Ding, X., and Yang, X. Iterative geometry encoding volume for stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21919–21928, 2023.

Xu, H. and Zhang, J. Aanet: Adaptive aggregation network for efficient stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1959–1968, 2020.

Yamaguchi, K., McAllester, D., and Urtasun, R. Efficient joint segmentation, occlusion labeling, stereo and flow estimation. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pp. 756–771. Springer, 2014.

Yang, J., Alvarez, J. M., and Liu, M. Non-parametric depth distribution modelling based depth inference for multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8626–8634, 2022.

Zbontar, J. and LeCun, Y. Computing the stereo matching cost with a convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

Zhang, F., Prisacariu, V., Yang, R., and Torr, P. H. Ga-net: Guided aggregation net for end-to-end stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 185–194, 2019.

Zhang, F., Qi, X., Yang, R., Prisacariu, V., Wah, B., and Torr, P. Domain-invariant stereo matching networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pp. 420–439. Springer, 2020.

Zhang, J., Wang, X., Bai, X., Wang, C., Huang, L., Chen, Y., Gu, L., Zhou, J., Harada, T., and Hancock, E. R. Revisiting domain generalized stereo matching networks from a feature consistency perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13001–13011, 2022.

Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2881–2890, 2017.

Zhao, H., Zhou, H., Zhang, Y., Chen, J., Yang, Y., and Zhao, Y. High-frequency stereo matching network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1327–1336, 2023.

# A. Training Details

We train our network on SceneFlow. The weight is initialized randomly. We use AdamW optimizer (Loshchilov & Hutter, 2019) with weight decay $10^{-5}$. The learning rate decreases from $2 \times 10^{-4}$ to $2 \times 10^{-8}$ according to the one cycle learning rate policy. We train the network for $2 \times 10^{5}$ iterations. The images will be randomly cropped to $320 \times 736$. For KITTI 2012 & 2015 benchmarks, we further fine tune the network on the training image pairs for $2.5 \times 10^{3}$ iterations. The learning rate starts from $5 \times 10^{-5}$ to $5 \times 10^{-9}$.

Following RAFT-Stereo (Lipson et al., 2021), we apply various image augmentations during training to avoid the over-fitting problem. Specifically, the augmentations include *(a)* color transformation, *(b)* occlusion, and *(c)* spatial transformation. In *(a)* color transformation, we randomly change the brightness, contrast, saturation and hue of the left and right images independently. The brightness and contrast factors are uniformly chosen from [0.6, 1.4]. The saturation factor is uniformly chosen from [0.0, 1.4]. The hue factor is uniformly chosen from [-0.16, 0.16]. In *(b)* occlusion, we randomly select a few rectangular regions in the right image, and set the pixels inside the regions as the mean color of the right image. The number of regions is chosen from $\{0, 1, 2, 3\}$ with probabilities $\{0.5, 0.166, 0.166, 0.166\}$. The position of the region is uniformly chosen in the right image, and the width and height are uniformly chosen from [50, 100]. In *(c)* spatial transformation, we randomly crop the left and right images to the resolution 320×736.

# B. Network Structure Details

In this part, we present more details for the *(i)* feature extraction and *(ii)* cost aggregation.

***(i)* Feature Extraction.** Given an input image, the module aims to output multi-scale 2D feature maps. More specifically, we first use a ResNet (He et al., 2016) to extract 2D feature maps of resolution 1/4 and 1/2 with respect to the input image. The ResNet contains 4 stages of non-linear transformation with 3, 16, 3, 3 residual blocks respectively, where each block is composed of convolutional layers and skip connections. And the spatial resolution is downsampled before the beginning of the first and third stages of transformation. Then we apply the spatial pyramid pooling (Zhao et al., 2017) on the 1/4-resolution feature map from the fourth stage of transformation to enlarge the receptive field. In the end, we upsample the enhanced feature map from 1/4 to 1/2 and fuse it with the 1/2-resolution feature map from the first stage of transformation in ResNet. The final outputs are the feature maps of 1/4 and 1/2 resolution. We apply the same network and weights to extract features from left and right images. The details of the network structure and the resolution of the feature maps are shown in Tab.(9).

***(ii)* Cost Aggregation.** We use the stacked hourglass architecture (Newell et al., 2016) to transform the stereo cost volume and aggregate the matching cost. For the coarse and refined stages, the structures are the same except for the number of feature channels. Specifically, the network consists of three 3D hourglasss as in Chang & Chen (2018). Each hourglass first downsamples the volume hierarchically to 1/2 and 1/4 resolution with respect to the input volume, and then upsamples in sequence to recover the resolution. The above procedure helps aggregate the matching information across various scales. The final output is a volume that represents the discrete distribution of disparity hypotheses. We present the details of a single hourglass structure in Tab.(10). For an input image with resolution $h \times w$, the $D$, $H$, $W$, $C$ are 192, $h/4$, $w/4$, 32 respectively in the coarse stage. In the refined stage, we set $D$, $H$, $W$, $C$ to be 16, $h/2$, $w/2$, 16 respectively.

# C. Experiments

## C.1. Ablation Study for Tolerance

In this part, we change the value of the tolerance $\tau$ in the binary search algorithm and observe its effects. As shown in Tab.(11), when decreasing the value of $\tau$, the search algorithm will iterate for more times to search for the optimal solution. And the error of the predicted disparity is reduced. When $\tau \geq 0.1$, the algorithm achieves the best accuracy.

## C.2. Ablation Studies for Huber Loss

In this part, we evaluate the effects of different loss functions. In Tab.(12), we evaluate the $L^2$ loss, the $L^1$ loss, and the Huber loss, i.e. a combination of $L^1$ and $L^2$ norm depending on the thresholding value $\beta$. The table clearly shows the benefit of using risk minimization loss under $L^1$.

*Table 9.* Network structure for feature extraction.

| Name | Layer Setting | Output Dimension |
|---|---|---|
| | ResNet | |
| Input | | $H \times W \times 3$ |
| Stem-1 | $3 \times 3, 32$ | $H \times W \times 32$ |
| Stem-2 | $3 \times 3, 32$ | $H \times W \times 32$ |
| Stem-3 | $3 \times 3, 32$ | $\frac{1}{2}H \times \frac{1}{2}W \times 32$ |
| Stage-1 | $\begin{bmatrix} 3 \times 3, 32 \\ 3 \times 3, 32 \end{bmatrix} \times 3$ | $\frac{1}{2}H \times \frac{1}{2}W \times 32$ |
| Stage-2 | $\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 16$ | $\frac{1}{4}H \times \frac{1}{4}W \times 64$ |
| Stage-3 | $\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 3$ | $\frac{1}{4}H \times \frac{1}{4}W \times 128$ |
| Stage-4 | $\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 3, \texttt{dila} = 2$ | $\frac{1}{4}H \times \frac{1}{4}W \times 128$ |
| | Spatial Pyramid Pooling | |
| Branch-1 | $64 \times 64$ avg pool<br>$3 \times 3, 32$<br>bilinear interpolation | $\frac{1}{4}H \times \frac{1}{4}W \times 32$ |
| Branch-2 | $32 \times 32$ avg pool<br>$3 \times 3, 32$<br>bilinear interpolation | $\frac{1}{4}H \times \frac{1}{4}W \times 32$ |
| Branch-3 | $16 \times 16$ avg pool<br>$3 \times 3, 32$<br>bilinear interpolation | $\frac{1}{4}H \times \frac{1}{4}W \times 32$ |
| Branch-4 | $8 \times 8$ avg pool<br>$3 \times 3, 32$<br>bilinear interpolation | $\frac{1}{4}H \times \frac{1}{4}W \times 32$ |
| | Concat [Stage-2, Stage-4, Branch-1, Branch-2, Branch-3, Branch-4] | $\frac{1}{4}H \times \frac{1}{4}W \times 32$ |
| Fusion-1 | $3 \times 3, 128$<br>$1 \times 1, 32$ | $\frac{1}{4}H \times \frac{1}{4}W \times 32$ |
| | UpSample | |
| Up-1 | nearest interpolation | $\frac{1}{2}H \times \frac{1}{2}W \times 32$ |
| | Add [Stage-1, Up-0] | $\frac{1}{2}H \times \frac{1}{2}W \times 32$ |
| Fusion-2 | $3 \times 3, 16$ | $\frac{1}{2}H \times \frac{1}{2}W \times 16$ |

*Table 10.* Network structure for 3D hourglass.

| Name | Layer Setting | Output Dimension |
|---|---|---|
| Input | | $D \times H \times W \times C$ |
| Conv-1 | $3 \times 3 \times 3, 2C$ | $\frac{1}{2}D \times \frac{1}{2}H \times \frac{1}{2}W \times 2C$ |
| Conv-2 | $3 \times 3 \times 3, 2C$ | $\frac{1}{2}D \times \frac{1}{2}H \times \frac{1}{2}W \times 2C$ |
| Conv-3 | $3 \times 3 \times 3, 4C$ | $\frac{1}{4}D \times \frac{1}{4}H \times \frac{1}{4}W \times 4C$ |
| Conv-4 | $3 \times 3 \times 3, 4C$ | $\frac{1}{4}D \times \frac{1}{4}H \times \frac{1}{4}W \times 4C$ |
| Atte-4 | $3 \times 3 \times 3, C$<br>$3 \times 3 \times 3, 4C$<br>sigmoid<br>prod Conv-4 | $\frac{1}{4}D \times \frac{1}{4}H \times \frac{1}{4}W \times 4C$ |
| Conv-5 | deconv $3 \times 3 \times 3, 2C$<br>add Conv-2 | $\frac{1}{2}D \times \frac{1}{2}H \times \frac{1}{2}W \times 2C$ |
| Atte-5 | $3 \times 3 \times 3, C$<br>$3 \times 3 \times 3, 2C$<br>sigmoid<br>prod Conv-5 | $\frac{1}{2}D \times \frac{1}{2}H \times \frac{1}{2}W \times 2C$ |
| Conv-6 | deconv $3 \times 3 \times 3, C$<br>add Input | $D \times H \times W \times C$ |
| Atte-6 | $3 \times 3 \times 3, C$<br>$3 \times 3 \times 3, C$<br>sigmoid<br>prod Conv-6 | $D \times H \times W \times C$ |

*Table 11.* Ablation studies for tolerance $\tau$ on Middlebury training set of quarter resolution. The first and second bests are in red and blue respectively. **Our method** in bold. All settings are trained on SceneFlow and evaluated on Middlebury training set without fine-tuning.

| Tolerance $\tau$ | Number of Iterations | $>$ 1px | | $>$ 2px | |
|---|---|---|---|---|---|
| | | Noc | All | Noc | All |
| 0.3 | 9 | 9.36 | 12.67 | 4.50 | 6.71 |
| **0.1** | **11** | **9.32** | **12.63** | **4.49** | **6.70** |
| 0.01 | 14 | 9.32 | 12.63 | 4.49 | 6.70 |

*Table 12.* Ablation studies for loss function on Middlebury training set of quarter resolution. The first and second bests are in red and blue respectively. **Our method** in bold. All settings are trained on SceneFlow and evaluated on Middlebury training set without fine-tuning.

| Loss | $>$ 1px | | $>$ 2px | |
|---|---|---|---|---|
| | Noc | All | Noc | All |
| $L^2$ | 9.83 | 13.19 | 4.79 | 7.06 |
| $\beta = 10.0$ | 9.41 | 12.73 | 4.55 | 6.76 |
| $\beta = 4.0$ | 9.36 | 12.68 | 4.51 | 6.72 |
| $\beta = 1.0$ | 9.33 | 12.64 | 4.50 | 6.70 |
| $\boldsymbol{L^1}$ | **9.32** | **12.63** | **4.49** | **6.70** |

### C.3. Ablation Studies for Network Architectures

In this part, we apply our method to the IGEV (Xu et al., 2023) framework. Specifically, we use our method to compute the initial disparities from the geometry encoding volume. The results are shown in Tab.(13). Our method improves the accuracy of IGEV.

*Table 13.* Ablation studies for IGEV on Middlebury training set of quarter resolution. The first and second bests are in red and blue respectively. **Our method** in bold. All methods are trained on SceneFlow and evaluated on Middlebury training set without fine-tuning.

| Backbone | Training | Test | Param (M) | Time (s) | $>$ 3px | | $>$ 4px | |
|---|---|---|---|---|---|---|---|---|
| | | | | | Noc | All | Noc | All |
| IGEV (Xu et al., 2023) | Expectation | Expectation | 12.60 | 0.34 | 4.47 | 6.64 | 3.46 | 5.32 |
| | **Expectation** | **L1-Risk** | **12.60** | **0.38** | **4.37** | **6.63** | **3.40** | **5.32** |

### C.4. Ablation Studies for Interpolation Kernel

In this part, we change the interpolation kernel from Laplacian to Gaussian and observe the effects. As shown in Tab.(14), we find the Laplacian kernel has better accuracy.

### C.5. Cross-Domain Generalization

In this part, we apply our method to ITSA (Chuah et al., 2022) only at inference time. We use the pre-trained model provided by ITSA, which is trained on synthetic images. As shown in Tab.(15), when evaluated on real-world datasets, our method can improve the performance on various networks and benchmarks.

### C.6. Ablation Studies for Training Using $L^1$ Risk

In this part, we provide more results on Middlebury using L1-risk minimization both at training and test time on several popular stereo-matching network architectures, demonstrating the usefulness and completeness of our approach to stereo matching problem. The results are shown in Tab.(16).

## D. Qualitative Results

In this section, we present more qualitative results on real-world datasets in Fig. 4, Fig. 5 and Fig. 6. It can be observed that in general our method generalizes and predicts high-frequency details better than other recent methods.

## E. Solution for Squared $L^2$ Norm Loss

In this part, we present the optimal solution when using the squared $L^2$ norm loss, i.e., $\mathcal{L}(y, x) = (y - x)^2$.

*Table 14.* Ablation studies for interpolation kernel on Middlebury training set of quarter resolution. The first and second bests are in red and blue respectively. **Our method** in bold. All settings are trained on SceneFlow and evaluated on Middlebury training set without fine-tuning.

| Kernel | Param (M) | Time (s) | $>$ 1px | | $>$ 2px | |
|---|---|---|---|---|---|---|
| | | | Noc | All | Noc | All |
| Gaussian | 11.96 | 0.25 | 9.35 | 12.66 | 4.50 | 6.71 |
| **Laplacian** | **11.96** | **0.25** | **9.32** | **12.63** | **4.49** | **6.70** |

*Table 15.* Cross-domain evaluation with ITSA. The first and second bests are in red and blue respectively. All methods are trained on SceneFlow and evaluated on Middlebury training set without fine-tuning.

| Backbone | Training | Test | KITTI 2012 | KITTI 2015 | Middlebury | ETH3D |
|---|---|---|---|---|---|---|
| ITSA-PSMNet | Expectation | Expectation | 5.2 | 5.8 | 9.6 | 9.8 |
| | Expectation | L1-Risk | 5.0 | 5.6 | 9.0 | 9.7 |
| ITSA-GwcNet | Expectation | Expectation | 4.9 | 5.4 | 9.3 | 7.1 |
| | Expectation | L1-Risk | 4.6 | 5.2 | 8.8 | 7.1 |
| ITSA-CFNet | Expectation | Expectation | 4.2 | 4.7 | 8.5 | 5.1 |
| | Expectation | L1-Risk | 4.1 | 4.7 | 8.4 | 5.0 |

$$\texttt{argmin}_y F(y, \mathbf{p}^m) = \texttt{argmin}_y \int (y-x)^2 p(x;\mathbf{p}^m)dx. \tag{11}$$

Firstly, we found the function $F(y, \mathbf{p}^m) = \int (y-x)^2 p(x;\mathbf{p}^m)dx$ is convex with respect to $y$, because

$$\int (\lambda y_1 + (1-\lambda)y_2 - x)^2 p(x;\mathbf{p}^m)dx \leq \int (\lambda(y_1-x)^2 + (1-\lambda)(y_2-x)^2)p(x;\mathbf{p}^m)dx \tag{12}$$

$$= \lambda \int (y_1-x)^2 p(x;\mathbf{p}^m)dx + (1-\lambda)\int (y_2-x)^2 p(x;\mathbf{p}^m)dx \tag{13}$$

Secondly, the optimal solution for the function $F(y, \mathbf{p}^m)$ can be obtained where $\partial F/\partial y = 0$, i.e.,

$$\frac{\partial F(y,\mathbf{p}^m)}{\partial y} = 2\int (y-x)p(x;\mathbf{p}^m)dx = 2y - 2\int xp(x;\mathbf{p}^m)dx = 0 \tag{14}$$

Therefore the optimal solution is $y = \int xp(x;\mathbf{p}^m)dx$.

## F. Evaluation Metrics

The definition of evaluation metrics (Geiger et al., 2012; Menze & Geiger, 2015) is below:
D1: Percentage of stereo disparity outliers in first frame.
BG: Percentage of outliers averaged only over background regions.
FG: Percentage of outliers averaged only over foreground regions.
ALL: Percentage of outliers averaged over all ground truth pixels.

*Table 16.* Ablation studies for $L^1$ risk on Middlebury training set of quarter resolution. The first and second bests are in red and blue respectively. **Our method** in bold. All methods are trained on SceneFlow and evaluated on Middlebury training set without fine-tuning.

| Backbone | Training | Test | $> 1$px |  | $> 2$px |  |
|---|---|---|---|---|---|---|
|  |  |  | Noc | All | Noc | All |
| PSMNet (Chang & Chen, 2018) | Expectation | Expectation | 15.42 | 21.01 | 7.53 | 12.17 |
|  | **L1-Risk** | **L1-Risk** | **15.27** | **20.67** | **7.48** | **11.92** |
| GCNet (Kendall et al., 2017a) | Expectation | Expectation | 19.93 | 25.72 | 11.15 | 16.12 |
|  | **L1-Risk** | **L1-Risk** | **16.31** | **22.19** | **8.55** | **13.45** |

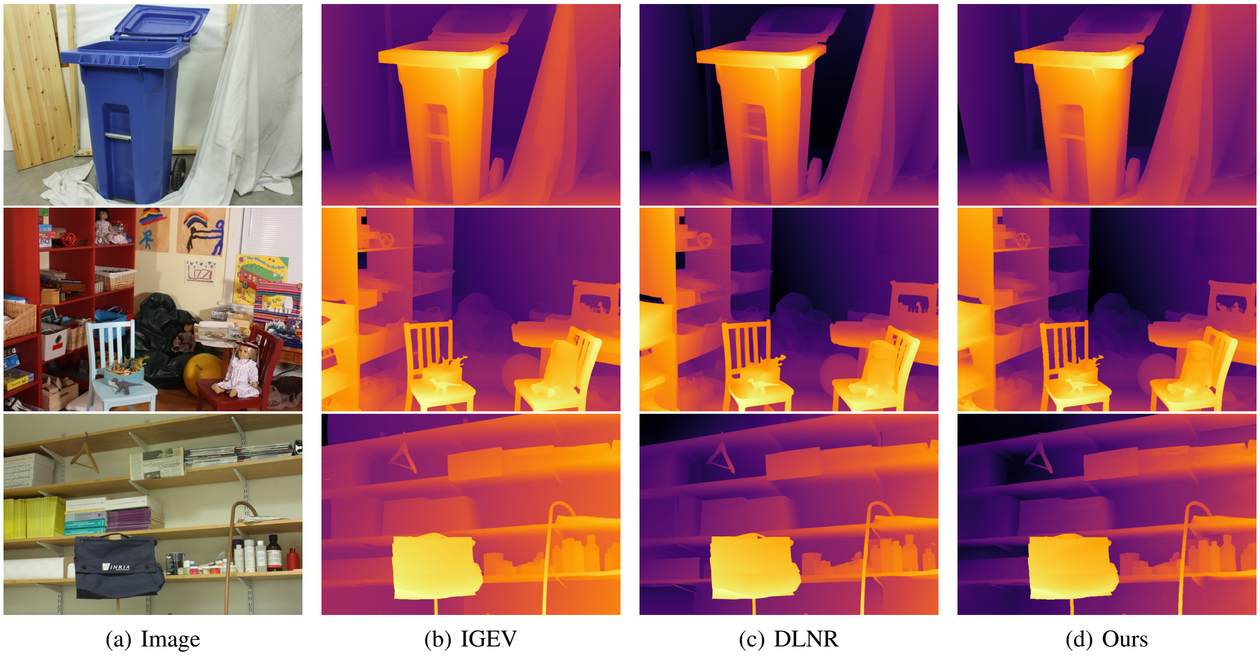


*Figure 4.* **Qualitative Comparison.** We compare our method with recent state-of-the-art methods such as IGEV (Xu et al., 2023), DLNR (Zhao et al., 2023) on Middlebury (Scharstein & Szeliski, 2002). All methods are trained only on SceneFlow (Mayer et al., 2016), and evaluated at quarter resolution.

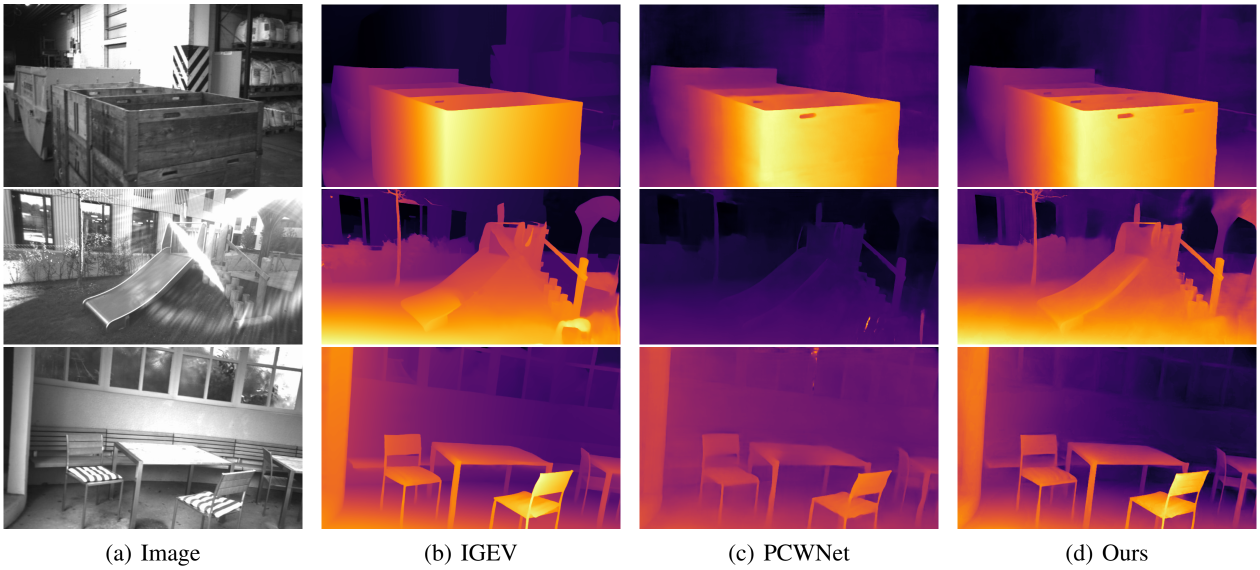


*Figure 5.* **Qualitative Comparison.** We compare our method with recent state-of-the-art methods such as IGEV (Xu et al., 2023), PCWNet (Shen et al., 2022) on ETH 3D (Schöps et al., 2017). All methods are trained only on SceneFlow (Mayer et al., 2016).

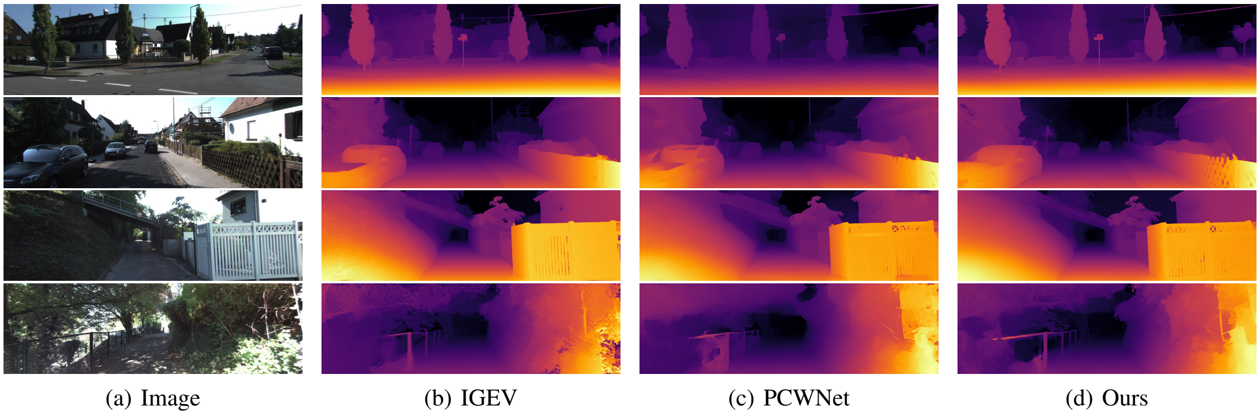


*Figure 6.* **Qualitative Comparison.** We compare our method with recent state-of-the-art methods such as IGEV (Xu et al., 2023), PCWNet (Shen et al., 2022) on KITTI 2012 (Geiger et al., 2012). All methods are trained only on SceneFlow (Mayer et al., 2016).