Structure-grounded Training Strategies Aid Generalization in Stereo Matching

Supplementary Material

A. Depth Score Calculation in Figure 1 of Main Text

For a fixed (model, dataset) and a given training strategy s, we combine endpoint error (EPE; lower is better) and the $<$ 3px accuracy (higher is better) via a normalized composite score:

$$\begin{aligned}\mathrm{DepthScore}_s = \alpha\left(1 - \frac{EPE_s - EPE_{\min}}{EPE_{\max} - EPE_{\min}}\right) \\ + (1-\alpha)\,\frac{\mathrm{Acc}_s - \mathrm{Acc}_{\min}}{\mathrm{Acc}_{\max} - \mathrm{Acc}_{\min}},\end{aligned} \tag{1}$$

where Acc denotes $<$3px(%), and the $\min$ / $\max$ operators are taken *across strategies within the same (model, dataset) pair*. We use $\alpha = \frac{1}{2}$ by default (equal weighting). When a denominator is zero (all strategies identical on that metric), we set the corresponding normalized term to 0. For visualization in radar plots, we apply a monotonic remapping $s' = \varepsilon + (1-\varepsilon)\,s$ with $\varepsilon = 0.15$ to avoid polygon degeneracy at the origin; this does *not* affect ranking. For bar plots, we report the dataset-averaged DepthScore per model and annotate Δ relative to the model's Baseline strategy.

B. Theoretical Analysis of Geometry-oriented Data Augmentation

B.1. Pixel-Level Correspondence as Bipartite Matching

We represent the task of matching pixels between two views as finding a maximum-weight matching in a complete bipartite graph. Let n be the number of feature points (pixels) we extract in each image. Denote by $\mathcal{I} = \{a_1, \ldots, a_n\}$ the set of source-image pixels and by $\mathcal{J} = \{b_1, \ldots, b_n\}$ the set of target-image pixels. We apply a feature extractor $f(\cdot)$ (e.g., a local descriptor) to each pixel, and a similarity function $s(\cdot,\cdot)$ (e.g., dot-product or cosine similarity) to pairs of descriptors. The resulting affinity between $a_i \in \mathcal{I}$ and $b_j \in \mathcal{J}$ can be denoted by

$$w_{ij} = s\big(f(a_i),\, f(b_j)\big)\,, \tag{2}$$

and we collect these into the weight matrix $W = [w_{ij}] \in \mathbb{R}^{n\times n}$.

We note that a matching is a bijection $\pi : \{1,\ldots,n\} \to \{1,\ldots,n\}$ that assigns each source pixel a_i to exactly one target pixel $b_{\pi(i)}$. Thus, we denote this permutation by π and score it by the total affinity

$$S(\pi) \;=\; \sum_{i=1}^{n} w_{i,\pi(i)}\,. \tag{3}$$

The optimal correspondence π^* maximizes this total score:

$$\pi^* \;=\; \arg\max_{\pi} S(\pi)\,. \tag{4}$$

To measure how clearly π^* stands out from other candidates, we define the *matching gap* as

$$g \;=\; S(\pi^*) \;-\; \max_{\pi \neq \pi^*} S(\pi)\,, \tag{5}$$

where the second term is the highest score among all permutations other than π^*. A positive gap $g > 0$ indicates that π^* is the unique optimum.

B.2. Texture Augmentation as Random Perturbation

In regions with weak texture, many pairs (a_i, b_j) yield similar affinities, leading to small g and ambiguous matching. We increase surface texture by overlaying random line patterns on the object, which perturbs each affinity by an independent random variable. Let ϵ_{ij} be zero-mean noise with variance σ^2, modeling the effect of texture:

$$\widetilde{w}_{ij} = w_{ij} + \epsilon_{ij}, \quad \epsilon_{ij} \sim \mathcal{D}(0, \sigma^2), \tag{6}$$

and write $\widetilde{W} = [\widetilde{w}_{ij}]$. Under this perturbation, the score of any matching π becomes

$$\begin{aligned} S'(\pi) &= \sum_{i=1}^{n} \widetilde{w}_{i,\pi(i)} \\ &= \sum_{i=1}^{n} \big(w_{i,\pi(i)} + \epsilon_{i,\pi(i)}\big) \\ &= S(\pi) + E(\pi), \\ E(\pi) &= \sum_{i=1}^{n} \epsilon_{i,\pi(i)}, \end{aligned} \tag{7}$$

where $E(\pi)$ is the total noise contribution for matching π. For each non-optimal permutation $\pi \neq \pi^*$, define

$$\Delta(\pi) = S(\pi^*) - S(\pi), \quad X(\pi) = E(\pi^*) - E(\pi). \tag{8}$$

Hypothesis: A larger matching gap improves pixel-level matching.

Increasing the matching gap g through texture augmentation can contribute to reducing matching difficulties in three folds:

1. Robustness to descriptor errors. In practice, computed affinities w_{ij} suffer bounded errors from descriptor quantization, sensor noise, or illumination changes. Suppose each w_{ij} is perturbed by at most δ, yielding observed weights $\widehat{w}_{ij} \in [w_{ij}-\delta,\ w_{ij}+\delta]$. Then for the true matching π^*,

$$\widehat{S}(\pi^*) \geq \sum_{i=1}^{n}(w_{i,\pi^*(i)} - \delta) = S(\pi^*) - n\delta, \tag{9}$$

and for any competitor π,

$$\widehat{S}(\pi) \leq \sum_{i=1}^{n}(w_{i,\pi(i)} + \delta) = S(\pi) + n\delta. \tag{10}$$

Thus the perturbed gap satisfies

$$\widehat{g} = \widehat{S}(\pi^*) - \max_{\pi\neq\pi^*} \widehat{S}(\pi) \ \geq\ g - 2n\delta. \tag{11}$$

If the original gap g exceeds $2n\delta$, the optimum remains π^* despite these errors. Therefore, a larger g confers greater *tolerance to deterministic weight perturbations*.

2. Enhanced pruning in matching algorithms. Combinatorial solvers (e.g. Hungarian method, auction algorithm) build partial matchings of size p with score S_p and estimate the maximum possible completion by adding at most $(n-p)\max_{i,j} w_{ij}$. A branch is pruned if

$$S_p + (n-p)\max_{i,j} w_{ij} < S(\pi^*) - g. \tag{12}$$

Larger g allows for *more aggressive pruning* of suboptimal branches, reducing the algorithm's search space and runtime.

3. Exponential decay of misassignment probability. Under random perturbation, π^* remains optimal if for every $\pi \neq \pi^*$,

$$\Delta(\pi) + X(\pi) > 0. \tag{13}$$

Since $X(\pi)$ has zero mean and variance $\mathrm{Var}[X(\pi)] = O(n\sigma^2)$, Hoeffding's inequality yields

$$\Pr\big[X(\pi) < -\Delta(\pi)\big] \leq \exp\Big(-\frac{\Delta(\pi)^2}{2\,n\,\sigma^2}\Big). \tag{14}$$

A union bound over all $n! - 1$ competitors gives

$$\Pr\big[\pi^* \text{ remains optimal}\big] \geq 1 - (n!-1)\exp\Big(-\frac{g^2}{2\,n\,\sigma^2}\Big). \tag{15}$$

Thus the probability of misassignment decays *exponentially* in $g^2/(n\sigma^2)$, showing that a larger gap substantially *reduces the risk of incorrect matches* under random texture noise.

C. Additional Results of Geometry-oriented Augmentation

Our augmentation preserves stereo consistency while introducing *two valid correspondences* for pixels within the perturbed regions: (i) the original correspondence $\mathbf{p}_r^o$, and (ii) the synthetic correspondence $\mathbf{p}_r^a$ induced by the warped disparity. As illustrated in Fig. S1, a pixel in the left image may simultaneously admit both $\mathbf{p}_r^o$ and $\mathbf{p}_r^a$ on the right view. The original correspondence $\mathbf{p}_r^o$ aligns with semantic priors and reflects the original disparity, while the synthetic correspondence $\mathbf{p}_r^a$ stems from the augmented disparity. In training, we supervise using $\mathbf{p}_r^a$, encouraging the network to prioritize fine-grained geometric cues over semantic bias. This is particularly beneficial in planar textured regions, where semantic guidance alone may hallucinate depth from texture, while geometric cues enforce the correct planar depth.

Further examples of randomly generated geometry-oriented augmentations are provided in Figures S2 and S3.

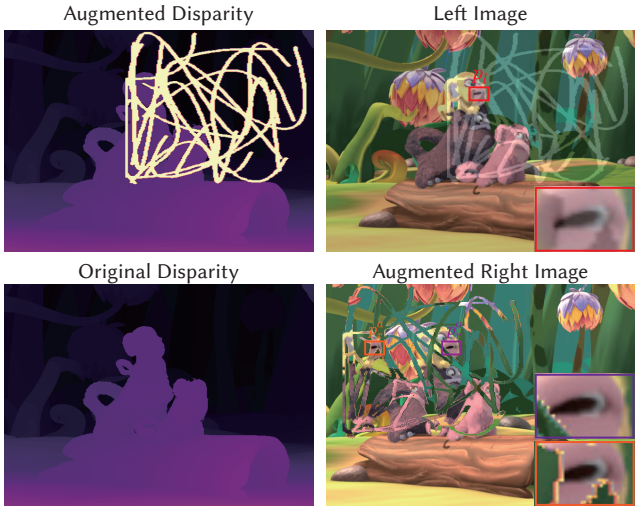

Figure S1. Illustration of dual correspondences induced by augmentation. A selected pixel $\mathbf{p}_l$ in the left image corresponds to the original match $\mathbf{p}_r^o$ in the right image (refer to the original disparity) and to the synthetic match $\mathbf{p}_r^a$ created by the augmented disparity.

D. Additional Training on Released Models

We further apply our training strategies to the official SceneFlow pretrained models released by RAFT-Stereo, IGEV-Stereo, and Selective-IGEV. Starting from the published checkpoints, we continue training on SceneFlow using our proposed training strategies, with identical hyperparameters across models (batch size 4, 100k steps, learning rate 1×10^{-5}; for Selective-IGEV, batch size 3 and 135k steps due to memory). For fair comparison, the *Baseline* results in Tables S1a–S1c are obtained by continuing training for the same number of steps using the official training scheme

Figure S2. Examples of geometry-oriented data augmentation with Ribbon mode.

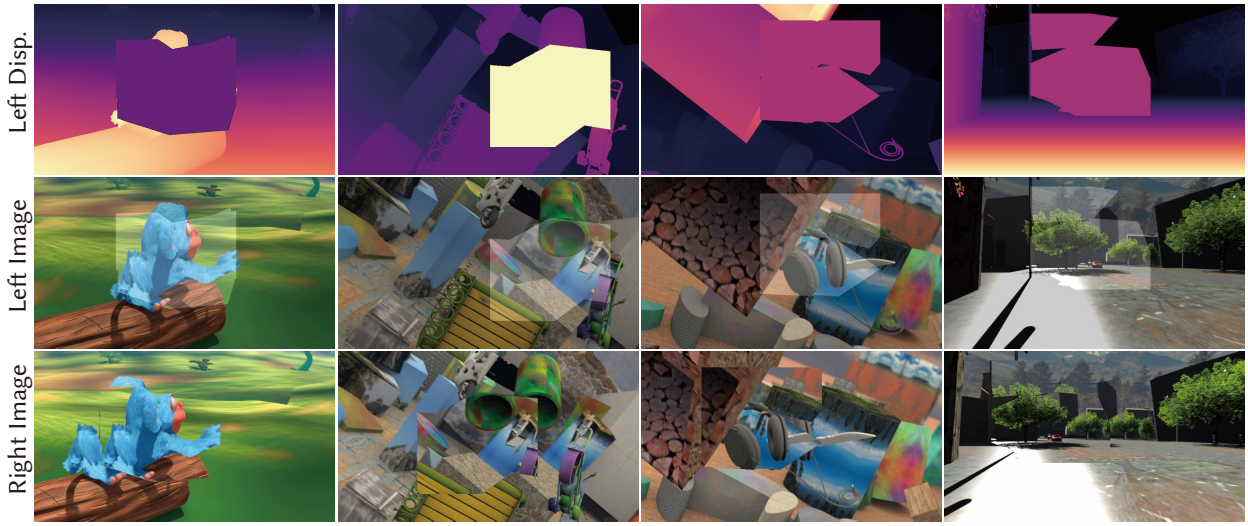

Figure S3. Examples of geometry-oriented data augmentation with Blob mode.

from each paper. We then evaluate the resulting models on KITTI 2012/2015, Middlebury, and ETH3D.

As shown in Tables S1a–S1c, our strategies consistently improve or maintain accuracy compared with the baselines. These results demonstrate that our training paradigm can be applied to existing SceneFlow-trained models and leads to measurable performance improvements across diverse architectures.

E. Additional Qualitative Results

Figures S4, S6, S5, and S7 extend the visual result of Fig. 8 in the main paper to all four models discussed there (RAFT-Stereo, IGEV-Stereo, Selective-IGEV, and DLNR). For each model, we show disparity predictions from the baseline and from our training strategies side by side.

Table S1. Results obtained by continuing training from the official SceneFlow-pretrained models of RAFT-Stereo, IGEV-Stereo, and Selective-IGEV. **Bold** = best, underline = second best, yellow background = improved over Baseline.

(a) RAFT-Stereo

Method	KITTI-12		KITTI-15		Middlebury		ETH3D	
	EPE (px)	< 3 px (%)	EPE (px)	< 3 px (%)	EPE (px)	< 3 px (%)	EPE (px)	< 3 px (%)
Baseline	0.929	95.1	1.173	94.1	**1.015**	**94.1**	0.275	98.7
GeomAug	0.923	95.2	1.150	94.3	1.099	93.2	0.286	98.8
AuxTask	**0.905**	**95.3**	**1.136**	**94.4**	1.172	93.6	**0.256**	**99.0**
UpdReg	0.966	94.8	1.195	94.0	1.275	93.0	0.270	98.8
Joint	0.908	**95.3**	1.154	94.2	1.332	93.9	0.267	98.9

(b) IGEV-Stereo

Method	KITTI-12		KITTI-15		Middlebury		ETH3D	
	EPE (px)	< 3 px (%)	EPE (px)	< 3 px (%)	EPE (px)	< 3 px (%)	EPE (px)	< 3 px (%)
Baseline	1.076	94.2	1.199	94.0	**0.891**	94.0	0.529	98.3
GeomAug	1.048	94.4	1.192	93.9	1.006	**94.6**	0.335	**98.5**
AuxTask	1.109	94.0	1.202	93.9	1.044	94.1	**0.318**	**98.5**
UpdReg	1.150	93.5	1.252	93.5	0.989	94.5	0.540	98.2
Joint	**1.002**	**94.8**	**1.182**	**94.2**	1.033	94.5	0.383	98.3

(c) Selective-IGEV

Method	KITTI-12		KITTI-15		Middlebury		ETH3D	
	EPE (px)	< 3 px (%)	EPE (px)	< 3 px (%)	EPE (px)	< 3 px (%)	EPE (px)	< 3 px (%)
Baseline	1.069	93.9	1.265	93.9	0.879	**95.6**	0.354	98.3
GeomAug	1.052	**94.2**	1.253	**94.0**	**0.816**	94.9	0.373	98.3
AuxTask	1.095	93.7	1.241	93.9	0.844	94.5	0.384	98.2
UpdReg	1.076	93.8	1.255	93.8	0.860	**95.6**	0.353	98.3
Joint	**1.042**	**94.2**	**1.220**	**94.0**	0.820	**95.6**	**0.340**	**98.5**

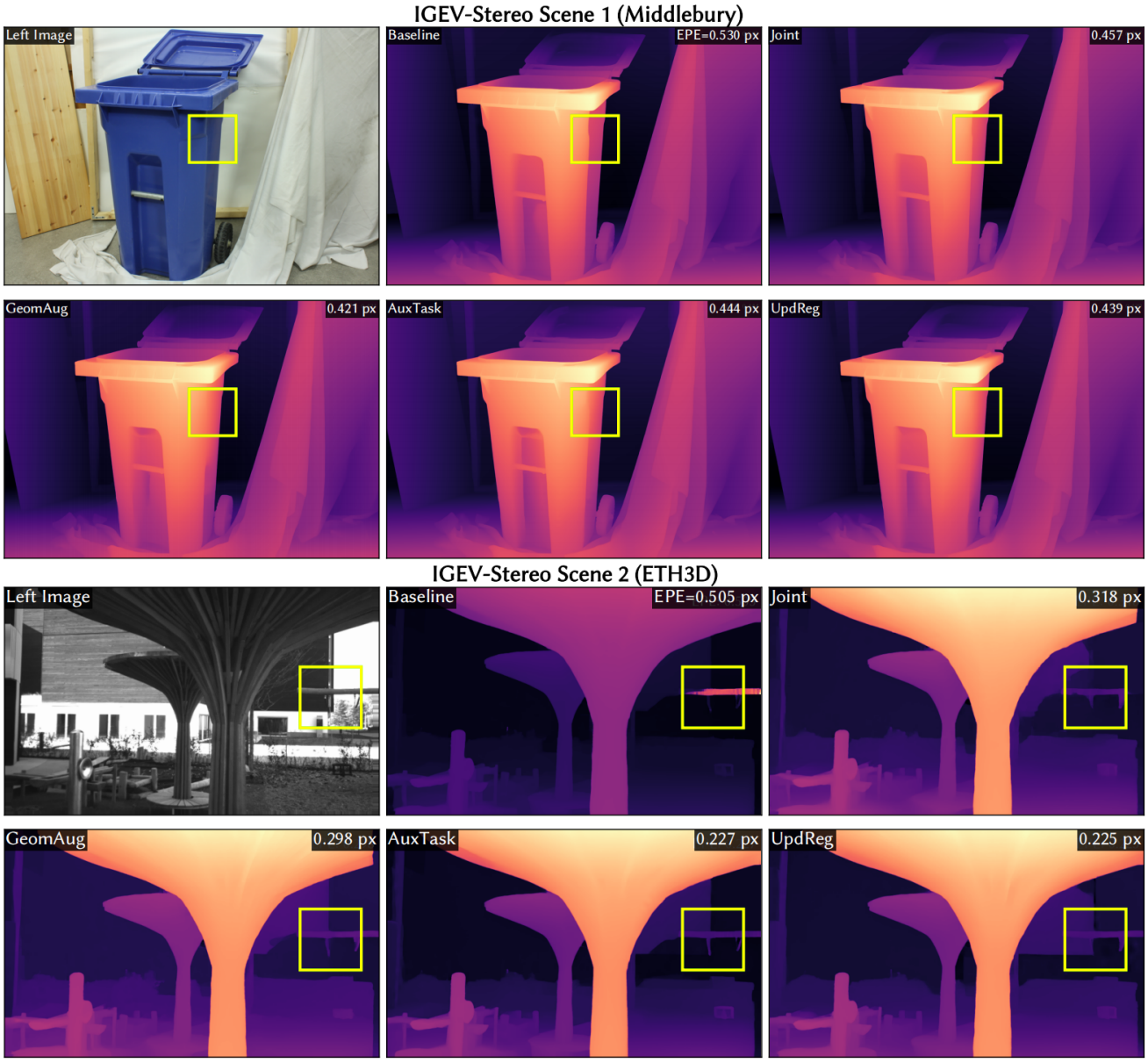

Figure S4. Qualitative comparison on two scenes (from KITTI 2012 and Middlebury) based on IGEV-Stereo. Two scenes are shown to compare our four training strategies with the baseline, original IGEV-Stereo. The highlighted boxes emphasize difficult areas (e.g., textureless regions, occlusions, and thin structures) where disparities diverge across methods. Per-panel EPE (top-right) in disparity map quantifies the error and complements the visual comparison, lower which is better. (top row: Left Image, Baseline, Joint; bottom row: GeomAug, AuxTask, UpdReg).

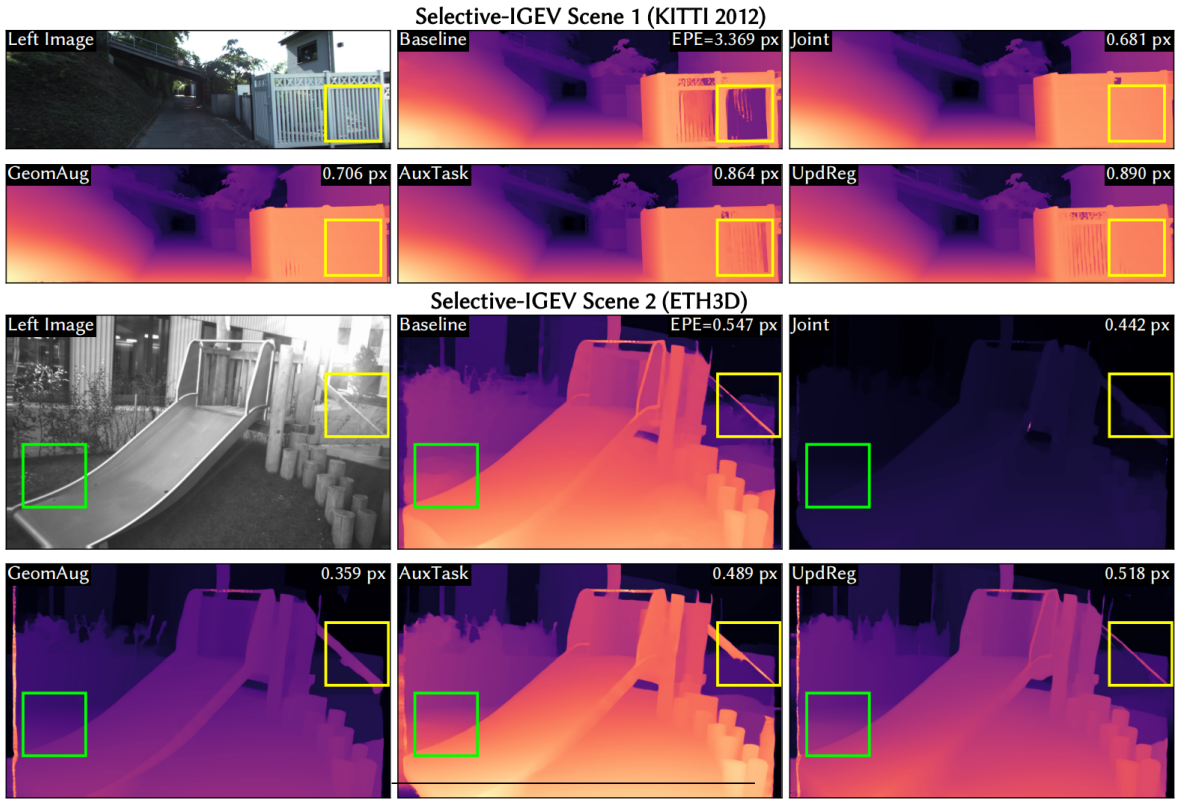

Figure S5. Qualitative comparison on two scenes (from KITTI 2012 and ETH3D) based on Selective-IGEV. Two scenes are shown to compare our four training strategies with the baseline, original Selective-IGEV.

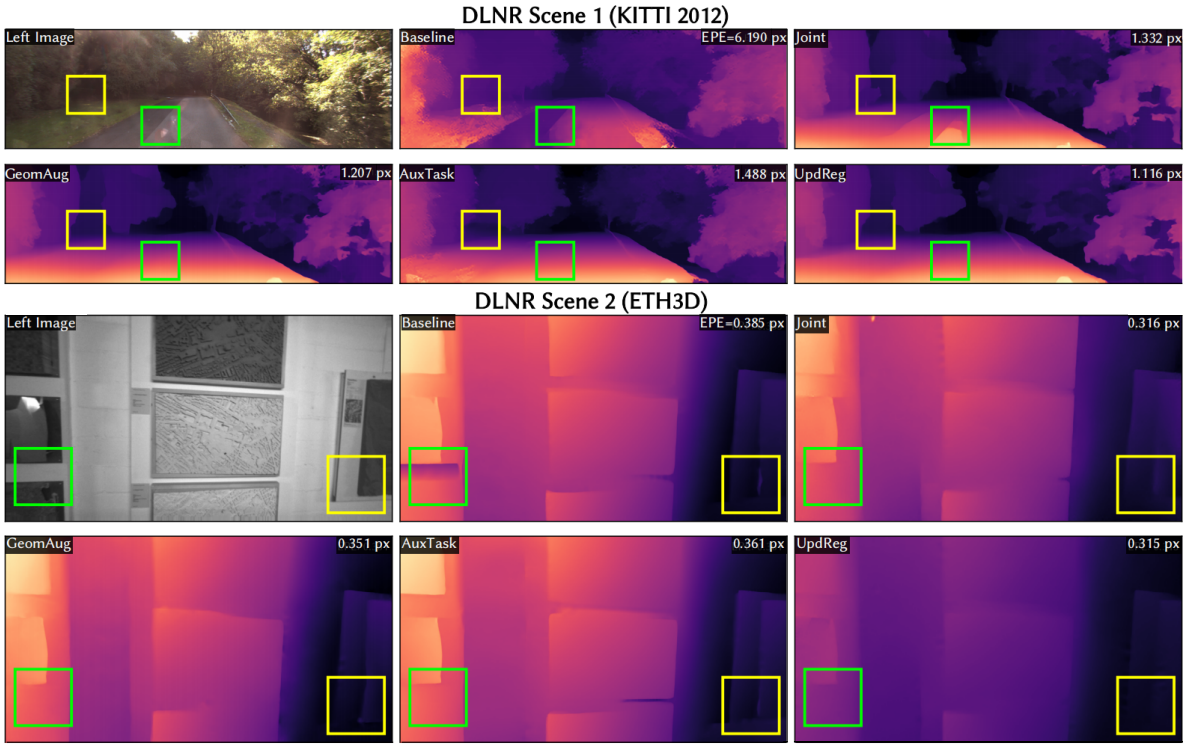

Figure S6. Qualitative comparison on two scenes (from KITTI 2012 and ETH3D) based on DLNR model. Two scenes are shown to compare our four training strategies with the baseline, original DLNR.

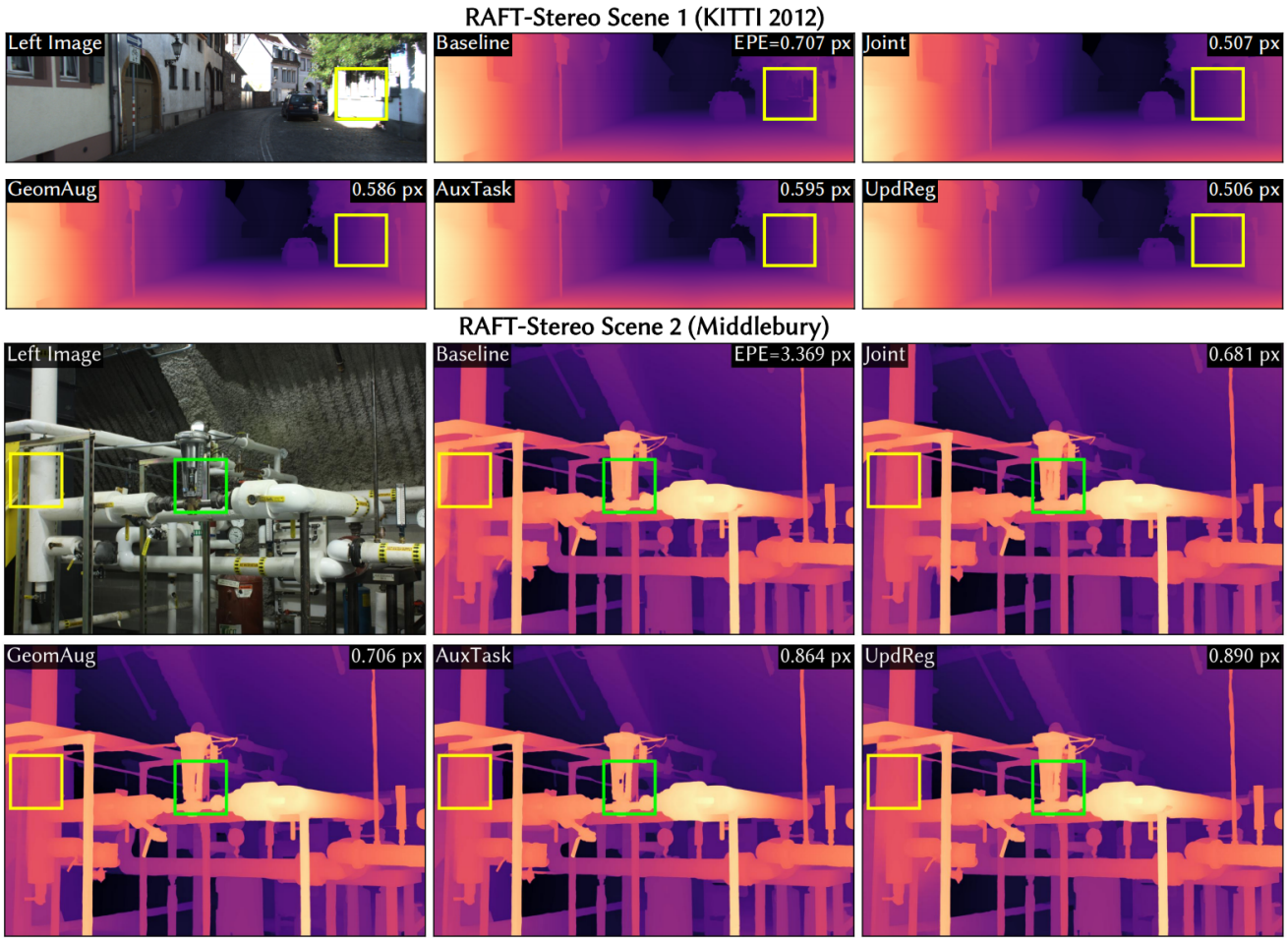

Figure S7. Qualitative comparison on two scenes (from KITTI 2012 and Middlebury) based on RAFT-Stereo. Two scenes are shown to compare our four training strategies with the baseline, original RAFT-Stereo. (top row: Left Image, Baseline, Joint; bottom row: GeomAug, AuxTask, UpdReg).