Revisiting Stereo Depth Estimation From a Sequence-to-Sequence Perspective with Transformers

Zhaoshuo Li, Xingtong Liu, Nathan Drenkow, Andy Ding, Francis X. Creighton, Russell H. Taylor, and Mathias Unberath

Johns Hopkins University
{zli122, mathias}@jhu.edu

Abstract

Stereo depth estimation relies on optimal correspondence matching between pixels on epipolar lines in the left and right images to infer depth. In this work, we revisit the problem from a sequence-to-sequence correspondence perspective to replace cost volume construction with dense pixel matching using position information and attention. This approach, named STereo TRansformer (STTR), has several advantages: It 1) relaxes the limitation of a fixed disparity range, 2) identifies occluded regions and provides confidence estimates, and 3) imposes uniqueness constraints during the matching process. We report promising results on both synthetic and real-world datasets and demonstrate that STTR generalizes across different domains, even without fine-tuning.

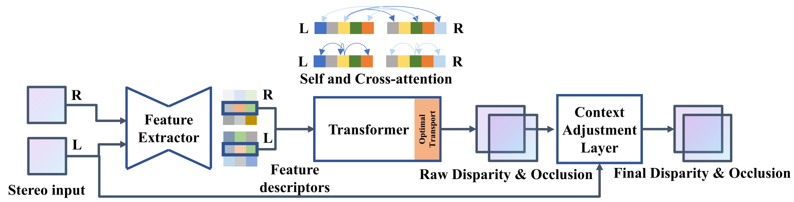
1. Introduction

Stereo depth estimation is of substantial interest since it enables the reconstruction of 3D information. To this end, corresponding pixels are matched between the left and right camera image; the difference in corresponding pixel location, i. e. the disparity, can then be used to infer depth and reconstruct the 3D scene. Recent deep learning-based approaches to stereo depth estimation have shown promising results but several challenges remain.

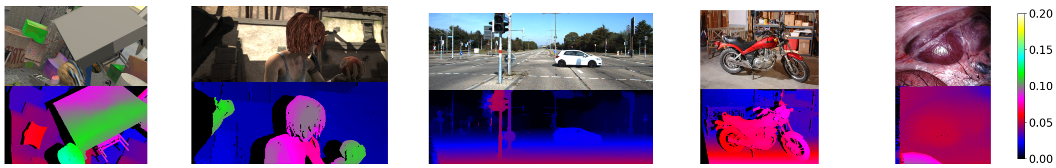
One such challenge relates to the use of a limited disparity range. Disparity values can, in theory, range from zero to the image width depending on the resolution/baseline of the cameras, and their proximity to the physical objects. However, many of the best performing approaches are constrained to a manually pre-specified disparity range (typically a maximum of 192 px) [21]. These methods rely on “cost volumes” in which matching costs are computed for multiple candidate matches and a final predicted disparity value is computed as the aggregated sum. This self-imposed disparity range is necessary to enable memory-feasible implementations of these methods but is not flexible to properties of the physical scene and/or the camera setup. In applications such as autonomous driving and endoscopic intervention, it is important to recognize close objects irrespective of camera setup (with disparity values potentially larger than 192) to avoid collisions, suggesting the need to relax the fixed disparity range assumption.

Geometric properties and constraints such as occlusion and matching uniqueness, which led to the success of non-learning based approaches such as [18], are also often missing from learning-based approaches. For stereo depth estimation, occluded regions do not have a valid disparity. Prior algorithms generally infer disparities for occluded regions via a piece-wise smoothness assumption, which may not always be valid. Providing a confidence estimate together with the disparity value would be advantageous for down-stream analysis, such as for registration or scene understanding algorithms, to enable weighting or rejection of occluded and low-confidence estimates. However, most prior approaches do not provide such information. Moreover, pixels in one image should not be matched to multiple pixels in the other image (up to image resolution) since they correspond to the same location in the physical scene [28]. Although this constraint can be clearly useful to resolve ambiguity, most existing learning-based approaches do not impose it.

The aforementioned problems largely arise from shortcomings of the contemporary view of stereo matching which attempts to construct a cost volume. Approaches that consider disparity estimation from a sequence-to-sequence matching perspective along epipolar lines can avoid these challenges. Such methods are not new, to our knowledge, the first attempt using dynamic programming was proposed in 1985 [28], where intra- and inter-epipolar line information is used together with a uniqueness constraint. How-



(a) Network overview



(b) Scene Flow (c) MPI Sintel (d) KITTI 2015 (e) Middlebury 2014 (f) SCARED

Figure 1. (a) STTR estimates disparity by first extracting features from stereo images using a shared feature extractor. The extracted feature descriptors are then used by a Transformer for dense self- and cross-attention computation, yielding a raw disparity estimate. A context adjustment layer further refines the disparity with information across epipolar lines conditioned on the left image for cross epipolar line optimality. (b-f) Inference of STTR trained only on synthetic Scene Flow dataset. Top row shows the left images. Bottom row shows predicted disparities. The color map used to visualize disparity is relative to the image width and is shown on the right. Black color indicates occlusion. Best viewed in color.

ever, it only used similarities between pixel intensities as matching criteria, which is inadequate beyond local matching, and thus restricted its performance. Recent advances in attention-based networks that capture long-range associations between feature descriptors prompt us to revisit this perspective. We take advantage of the recent Transformer architecture [36] proposed for language processing and recent advances in feature matching [31], and present a new end-to-end-trained stereo depth estimation network named STereo TRansformer (STTR). The main advantage of STTR is that it computes pixel-wise correlation densely and does not construct a fixed-disparity cost volume. Therefore, STTR can mitigate the drawbacks of most contemporary approaches that were detailed above with little to no compromises in performance. We report competitive performance on synthetic and real image benchmarks and demonstrate that STTR trained only on synthetic images also generalizes well to other domains without refinement.

We make the following technical advances to enable the realization of STTR:

- Instead of pixel-wise intensity correlation in traditional stereo depth estimation methods like [28], we adopt a Transformer with alternating self- and cross-attentions combined with the optimal transport theory previously demonstrated in sparse feature matching [31]. This design allows us to match pixels explicitly and densely while imposing a uniqueness constraint.
- We provide a relative pixel distance encoding to the feature descriptor and use a customized attention mechanism to define discriminative features during the matching process. This helps resolve ambiguity during the matching process.
- We devise a memory-feasible implementation of STTR which enables the training of the proposed model on conventional hardware. For seamless distribution and reproducibility, our code is available online[1] and only uses existing PyTorch functions [29].

2. Related Work

2.1. Stereo Depth Estimation

In general, the stereo depth estimation task involves two key steps (1) feature matching and (2) matching cost aggregation [21]. Traditionally, the task is solved by dynamic programming techniques where matchings are computed using pixel intensities and costs are aggregated either horizontally in 1D [28] or multi-directionally in 2D [18].

More recently, learning-based methods that match features by dot-product *correlation* have emerged. Early ones such as [6] compute feature similarities based on a patch of features and refine the matching using a Markov Random Field. Subsequent approaches [25, 22] take advantage of learning-based feature extractors and compute similarities between the feature descriptors for each pixel. [39] further advances the performance with cross-scale information aggregation.

Concurrently, networks such as [20, 5] build a 4D feature volume by concatenating features at different disparities and learn to compute/aggregate the matching cost by *3D convolutions*. In [43], additional semi-global and local

[1]Code is available at *https://github.com/mli0603/stereo-transformer*

cost aggregation layers are proposed to improve the performance. Following the same idea of computing matching cost by learnt 3D convolutions, other works [40, 7, 41, 15] attempt to enable high resolution inference, mitigate the memory constraint, and/or leverage richer context information via a multi-resolution approach.

Hybrid approaches like [16] have also emerged, which combine explicit correlation and 3D convolutions for matching and cost aggregation respectively. Other works follow different design concepts, such as [2] with a classification-based approach for disparity estimation.

However, none of the above prior work exploits the sequential nature and geometric properties of stereo matching, which led to the success of non-learning based works such as [28, 18]. Moreover, whether matching is computed by correlation or learnt by 3D convolutions, a maximum disparity is set to mitigate memory and computation demands in the above works. For each pixel, there is a fixed and finite set of discretized locations where a pixel can be mapped, thus generating a matching cost volume. For disparities beyond this pre-defined range, these approaches simply cannot infer the correct match. This limits the generalization of the networks across different scenes and stereo camera configurations. In addition, most learning-based approaches do not handle occlusion explicitly, even if the disparity in occluded regions can theoretically be arbitrary. Lastly, no explicit uniqueness constraint is imposed during the matching process, which can inhibit performance due to inconsistencies of matching.

2.2. Comparison of STTR to Previous Learning-based Stereo Paradigms

We use a convolutional neural network as the feature extractor that feeds into a Transformer to capture long-range associations between pixels. STTR exploits the sequential nature and geometric properties of stereo matching.

STTR vs. Correlation-based Networks: STTR imposes a uniqueness constraint during the matching process to resolve ambiguities. STTR also alternates between intra- and inter-image correlation operations, named self- and cross-attention, and updates the feature representations by considering both image context and position information.

STTR vs. 3D Convolution-based Networks: STTR explicitly and densely computes the correlation between pixels in the left and right images. Instead of using 3D convolutions to aggregate a cost volume, we first match along epipolar lines and then aggregate the information across epipolar lines via 2D convolutions.

2.3. Attention Mechanism and Transformer

Attention has already proven to be an effective tool in natural language processing [36]. Recently, attention-based architectures has found applications in computer vision tasks, such as image classification [11], object detection [4], panoptic segmentation [37], and homography estimation and visual localization [31], improving on the results of pure CNN architectures. This is likely because attention can capture long-range associations which is of particular importance for the work presented here. We adopt a Transformer to revisit the sequence-to-sequence stereo matching paradigm originally proposed in [28].

3. The Stereo Transformer Architecture

In the following sections, we denote the height and width of the rectified left and right pair of images as I_h and I_w. We denote the channel dimension of feature descriptors as C.

3.1. Feature Extractor

We use an hourglass-shaped architecture similar to [23], with the exception that the encoding path is modified with residual connections and spatial pyramid pooling modules [5] for more efficient global context acquisition. The decoding path consists of transposed convolution, dense-blocks [19], and a final convolution layer. The feature descriptors for each pixel, denoted as vector e_I of size C_e, encode both local and global context. The final feature map is at the same spatial resolution as the input image.

3.2. Transformer

An overview of the Transformer architecture used here is provided in Fig. 2. We adopt the alternating attention mechanism in [31]: *Self-attention* computes attention between pixels along the epipolar line in the same image, while *cross-attention* computes attention of pixels along corresponding epipolar lines in the left and right images. Details on both attention modules follows in Section 3.2.1. As shown in Fig. 2, we alternate between computing self- and cross-attention for $N-1$ layers. This alternating scheme keeps updating the feature descriptors based on the image context and relative position, as discussed in Section 3.2.2.

In the last cross-attention layer, we use the most attended pixel to estimate the raw disparity. We add operations exclusive to this layer, including *optimal transport* for compliance with the uniqueness constraint (Section 3.2.3) and an *attention mask* for search space reduction (Section 3.2.4).

3.2.1 Attention

Attention modules [36] compute the attention between a set of *query* vectors and *key* vectors using dot-product similarity, which is then used to weigh a set of *value* vectors.

We adopt multi-head attention, which increases the expressivity of the feature descriptor by splitting the channel dimension of feature descriptors C_e into N_h groups $C_h = C_e/N_h$, where C_h is the channel dimension of each

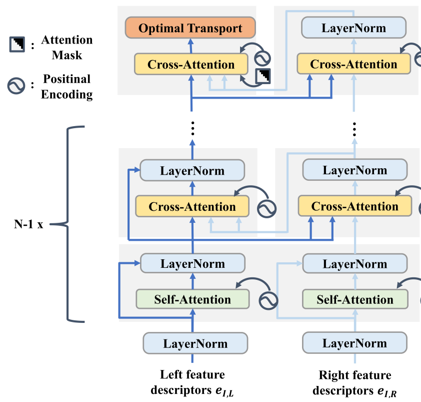


Figure 2. Overview of the Transformer module with alternating self- and cross-attention. Note that in the last cross-attention layer, the optimal transport and attention mask are added.

head and N_h is the number of heads. Therefore, each head can have different representations, and similarities can be computed per head. For each attention head h, a set of linear projections are used to compute the *query* vectors $\mathcal{Q}_h$, *key* vectors $\mathcal{K}_h$ and *value* vectors $\mathcal{V}_h$ using feature descriptors e_I as input:

$$\begin{aligned} \mathcal{Q}_h &= W_{\mathcal{Q}_h} e_I + b_{\mathcal{Q}_h} \\ \mathcal{K}_h &= W_{\mathcal{K}_h} e_I + b_{\mathcal{K}_h} \\ \mathcal{V}_h &= W_{\mathcal{V}_h} e_I + b_{\mathcal{V}_h} \, , \end{aligned} \tag{1}$$

where $W_{\mathcal{Q}_h}, W_{\mathcal{K}_h}, W_{\mathcal{V}_h} \in \mathbb{R}^{C_h \times C_h}$, $b_{\mathcal{Q}_h}, b_{\mathcal{K}_h}, b_{\mathcal{V}_h} \in \mathbb{R}^{C_h}$. We normalize the similarities via softmax and obtain α_h as

$$\alpha_h = \text{softmax}\left(\frac{\mathcal{Q}_h^T \mathcal{K}_h}{\sqrt{C_h}}\right) . \tag{2}$$

The output *value* vector $\mathcal{V}_\mathcal{O}$ can be computed as:

$$\mathcal{V}_\mathcal{O} = W_\mathcal{O} \, \text{Concat}(\alpha_1 \mathcal{V}_1, ..., \alpha_{N_h} \mathcal{V}_{N_h}) + b_\mathcal{O} \, , \tag{3}$$

where $W_\mathcal{O} \in \mathbb{R}^{C_e \times C_e}$ and $b_\mathcal{O} \in \mathbb{R}^{C_e}$. The output *value* vector $\mathcal{V}_\mathcal{O}$ is then added to the original feature descriptors to form a residual connection:

$$e_I = e_I + \mathcal{V}_\mathcal{O} \, . \tag{4}$$

For self-attention, the $\mathcal{Q}_h, \mathcal{K}_h, \mathcal{V}_h$ are computed from the same image. For cross-attention, $\mathcal{Q}_h$ is computed from the source image while $\mathcal{K}_h, \mathcal{V}_h$ are computed from the target image. Cross-attention is applied bi-directionally so in one case source→target is left→right and the other case is right→left.

3.2.2 Relative Positional Encoding

In large textureless areas, similarities between pixels can be ambiguous. This ambiguity, however, may be resolved by considering relative positional information with respect to prominent features, such as edges. Thus, we provide data-dependent spatial information via positional encoding e_p. We choose to encode relative pixel distances instead of absolute pixel locations due to its shift-invariance. In the vanilla Transformer [36], the absolute positional encoding e_p is directly added to the feature descriptor

$$e = e_I + e_p \, . \tag{5}$$

In that case, attention between the i-th and j-th pixel in Equation 2 can be expanded [10] (ignoring biases for simplicity) as

$$\begin{aligned} \alpha_{i,j} = & \underbrace{e_{I,i}^T W_\mathcal{Q}^T W_\mathcal{K} e_{I,j}}_{\text{(1) data-data}} + \underbrace{e_{I,i}^T W_\mathcal{Q}^T W_\mathcal{K} e_{p,j}}_{\text{(2) data-position}} + \\ & \underbrace{e_{p,i}^T W_\mathcal{Q}^T W_\mathcal{K} e_{I,j}}_{\text{(3) position-data}} + \underbrace{e_{p,i}^T W_\mathcal{Q}^T W_\mathcal{K} e_{p,j}}_{\text{(4) position-position}} \, . \end{aligned} \tag{6}$$

As shown, the term (4) entirely depends on position and should thus be left out, as disparity fundamentally depends on image content. Instead, we use relative positional encoding and remove the term (4) as

$$\begin{aligned} \alpha_{i,j} = & \underbrace{e_{I,i}^T W_\mathcal{Q}^T W_\mathcal{K} e_{I,j}}_{\text{(1) data-data}} + \\ & \underbrace{e_{I,i}^T W_\mathcal{Q}^T W_\mathcal{K} e_{p,i-j}}_{\text{(2) data-position}} + \underbrace{e_{p,i-j}^T W_\mathcal{Q}^T W_\mathcal{K} e_{I,j}}_{\text{(3) position-data}} \, , \end{aligned} \tag{7}$$

where $e_{p,i-j}$ denotes the positional encoding between the i-th and j-th pixel. Note that $e_{p,i-j} \neq e_{p,j-i}$. Intuitively, the attention depends on both content similarity and relative distance. Concurrent to our development, [17] found that a similar attention mechanism is beneficial for NLP tasks.

However, the computational cost of relative distance is quadratic in the image width I_w since for each pixel, there are I_w relative distances, and this computation needs to be done I_w times. We describe an efficient implementation which reduces the cost to linear. The details of the implementation follow in Appendix A.

3.2.3 Optimal Transport

Enforcing the uniqueness constraint of stereo matching was attempted in [28], where each pixel in the right image gets assigned to at most one pixel in the left image. However, this hard assignment prohibits gradient flow. By contrast, entropy-regularized optimal transport [9] is an ideal alternative due to its soft assignment and differentiability, and was previously demonstrated as beneficial for the related task of sparse feature [31] and semantic correspondence [24] matching. Given a cost matrix M of two marginal distributions a and b of length I_w, the entropy-regularized optimal transport attempts to find the optimal coupling matrix $\mathcal{T}$ by solving

$$\begin{aligned} \mathcal{T} = & \underset{\mathcal{T} \in R_+^{I_w \times I_w}}{\operatorname{argmin}} \sum_{i,j=1}^{I_w, I_w} \mathcal{T}_{ij} M_{ij} - \gamma E(\mathcal{T}) \\ & \text{s.t. } \mathcal{T} 1_{I_w} = a,\ \mathcal{T}^T 1_{I_w} = b \end{aligned} \tag{8}$$

where $E(\mathcal{T})$ is the entropy regularization. If the two marginal distributions a, b are uniform, $\mathcal{T}$ is also optimal for the assignment problem, which imposes a soft uniqueness constraint [30] and mitigates ambiguity [24]. The solution to Equation 8 can be found via the iterative Sinkhorn algorithm [9]. Intuitively, values in $\mathcal{T}$ represent the pairwise matching probabilities, similar to softmaxed attention in Equation 2. Due to occlusion, some pixels cannot be matched. Following [31], we augment the cost matrix by adding dustbins with a learnable parameter ϕ that, intuitively, represents the cost of setting a pixel unmatched.

In STTR, the cost matrix M is set to the negative of the attention computed by the cross-attention module in Equation 2, but without softmax, as optimal transport will normalize the attention values.

3.2.4 Attention Mask

Let x_L, x_R be the projected location of the same physical point onto the left and right epipolar lines respectively ($+x$ from left to the right). The spatial arrangement of the cameras in a stereo rig ensures that $x_R \leq x_L$ for all points after rectification. Therefore, in the last cross-attention layer, it is sufficient for each pixel in the left image to only attend to pixels that are further to the left of the same coordinate in the right image (i.e., attend only to points x in the right image where $x \leq x_L$). To impose such a constraint, we introduce a lower-triangular binary mask on the attention. Additional visualization can be found in Appendix B.

3.2.5 Raw Disparity and Occlusion Regression

In most prior work, a weighted-sum of all candidate disparity values is used. We instead regress disparity using a modified winner-take-all approach [35], which is robust against multi-modal distributions.

The raw disparity is computed by finding the location of the most probable match, denoted as k, from the optimal transport assignment matrix $\mathcal{T}$ and build a 3 px window $\mathcal{N}_3(k)$ around it. A re-normalization step is applied to the matching probabilities within the 3 px window such that the sum is 1. The weighted sum of the candidate disparities is the regressed raw disparity $\tilde{d}_{raw}(k)$. Denoting the matching probability in assignment matrix $\mathcal{T}$ to be t, we have

$$\tilde{t}_l = \frac{t_l}{\sum_{l \in \mathcal{N}_3(k)} t_l}, \text{ for } l \in \mathcal{N}_3(k) \tag{9}$$

$$\tilde{d}_{raw}(k) = \sum_{l \in \mathcal{N}_3(k)} d_l \tilde{t}_l \tag{10}$$

The sum of probabilities within this 3 px window represents an estimate of the confidence of the network with the current assignment, in the form of an inverse occlusion probability. Therefore, we can regress the occlusion probability $p_{occ}(k)$ using the same information as

$$p_{occ}(k) = 1 - \sum_{l \in \mathcal{N}_3(k)} t_l \,. \tag{11}$$

3.3. Context Adjustment Layer

The raw disparity and occlusion maps are regressed over epipolar lines and thus lack context across multiple epipolar lines. To mitigate this, we use convolutions to adjust the estimated values conditioned on the input image with cross epipolar line information. The overview of the context adjustment layer is in Fig. 3.

The raw disparity and occlusion maps are first concatenated with the left image along the channel dimension. Two convolution blocks are used to aggregate the occlusion information, followed by ReLU. The final occlusion is estimated by a Sigmoid activation. Disparities are refined by residual blocks which expand the channel dimension before ReLU activation then restores it to the original channel dimension. The expansion before the ReLU is to encourage better information flow [42]. Raw disparity is repeatedly concatenated with the residual block for better conditioning. The final output of the residual blocks is added to the raw disparity via a long skip connection.

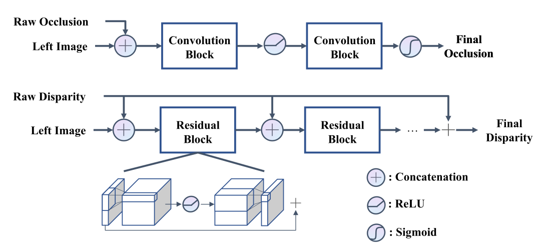


Figure 3. Overview of the context adjustment layer. Convolution blocks and Sigmoid activation are used for occlusion refinement (top) and residual blocks with long skip connections are used for disparity refinement (bottom). Qualitative result is in Appendix F.

3.4. Loss

We adopt the Relative Response loss L_{rr} proposed in [23] on the assignment matrix $\mathcal{T}$ for both sets of matched pixels $\mathcal{M}$ and sets of unmatched pixels $\mathcal{U}$ due to occlusion. The goal of the network is to maximize the attention on the true target location. Since disparity is subpixel, we use linear interpolation between the nearest integer pixels to find the matching probability t^*. Specifically, for the i-th pixel in the left image with ground truth disparity $d_{gt,i}$,

$$\begin{aligned} t_i^* &= \text{interp}(\mathcal{T}_i, p_i - d_{gt,i}) \\ L_{rr} &= \frac{1}{N_{\mathcal{M}}} \sum_{i \in \mathcal{M}} -log(t_i^*) + \frac{1}{N_{\mathcal{U}}} \sum_{i \in \mathcal{U}} -log(t_{i,\phi}) \end{aligned} \tag{12}$$

where *interp* denotes linear interpolation and $t_{i,\phi}$ is the unmatched probability. We use smooth L1 loss [13] on both raw and final disparities, denoted as $L_{d1,r}$ and $L_{d1,f}$. The final occlusion map is supervised via a binary-entropy loss $L_{be,f}$. The total loss is the summation:

$$L = w_1 L_{rr} + w_2 L_{d1,r} + w_3 L_{d1,f} + w_4 L_{be,f}\ , \tag{13}$$

where w are the loss weights.

3.5. Memory-Feasible Implementation

The memory consumption of the attention mechanism is quadratic in terms of the sequence length. Specifically, for a float32 precision computation,

$$\text{memory consumption in bits} = 32 I_h I_w^2 N_h N\ . \tag{14}$$

For example, given $I_w = 960$, $I_h = 540$ and $N_h = 8$, training a $N = 6$ layer Transformer consume approximately 216 GB, which is impractical on conventional hardware. Following [8], we adopt gradient checkpointing [14] for each self- and cross-attention layer, where the intermediate variables are not saved during the forward pass. During the backward pass, we run the forward pass again for the checkpointed layer to recompute the gradient. Thus, the memory consumption is bounded by the requirement of a single attention layer, which in theory enables the network to scale infinitely in terms of the number of attention layers N.

Moreover, we use mixed-precision training [27] for faster training speed and reduced memory consumption.

Lastly, we use an attention stride $s > 1$ to sparsely sample the feature descriptors, which is equivalent to a downsampling of the feature map.

Complexity Analysis: In existing cost-volume paradigms, correlation-based networks have a memory complexity of $\mathcal{O}(I_h I_w D)$, while 3D convolution-based networks have $\mathcal{O}(I_h I_w D C)$, where D is the maximum disparity value and C is the channel size. D is generally set to a fixed value less than I_w, sacrificing the ability to predict disparity values outside the range. STTR is of $\mathcal{O}(I_h I_w^2 / s^3)$, which offers an alternative trade-off where no maximum disparity is set. Given s, STTR runs at a *constant* memory consumption across *different* disparity ranges compared to prior work. During inference, s can be adjusted to a larger value which reduces memory consumption and maintains the maximal disparity range at the slight sacrifice of task performance. Quantitative analysis of the trade-off between performance and memory of s is in Appendix G. We also introduce a lightweight implementation of STTR without the flexibility to adjust s in Appendix H for faster speed and lower memory consumption. Comparison of inference speed/memory between STTR and prior work is in Appendix I.

4. Experiments, Results, and Discussion

Datasets: Scene Flow [25] FlyingThings3D subset is a synthetic dataset of random objects. MPI Sintel [3] is a synthetic dataset from animated film that contains various realistic artifacts such as specular reflections and motion blur. KITTI 2015 [26] is a street scene dataset. Middlebury 2014 [32] quarter resolution subset is an indoor scene dataset. SCARED [1] is a medical scene dataset of laparoscopic surgery. For pre-training, we use the default split of Scene Flow. For cross-domain generalization evaluation, we use all the provided data from each dataset. For KITTI 2015 benchmark evaluation, we train on KITTI 2012 and 2015 dataset and leave 20 images for validation. Details of the datasets and pre-processing step are in Appendix J. Training duration and number of parameters are in Appendix L.

Hyperparameters: In our experiments, we use 6 self- and cross-attention layers with $C_e = 128$. We run the Sinkhorn algorithm for 10 iterations. We use attention stride of $s = 3$ during training. We use AdamW as the optimizer with weight decay of 1e-4. We set all loss weights w to 1. We pre-train on Scene Flow for 15 epochs using a fixed learning rate of 1e-4 for feature extractor and Transformer, and 2e-4 for the context adjustment layer. To simulate realistic stereo artifacts, we use asymmetric (i. e. different for left and right images) augmentation, including RGB shift, Gaussian noise, brightness/contrast shift, vertical shift and rotation. For KITTI 2015 benchmark submission, we finetune the pre-trained model using the exponential learning rate scheduler with a decay of 0.99 for 400 epochs. We conduct our experiments on one Nvidia Titan RTX GPU. We use both 3 px Error (percentage of errors larger than 3 px) and EPE (absolute error) as evaluation metrics. Please note that all quantitative metrics reported in the remainder of this section refer to *non-occluded regions* only; we use IOU to evaluate *occlusion estimation*.

4.1. Ablation Studies

We conduct ablation studies using the Scene Flow synthetic dataset, and provide quantitative results for the effects of the attention mask, optimal transport layer, context adjustment layer and positional encoding summarized in Table 1. Following prior work, we validate on the test split directly since Scene Flow is only used for pre-training.

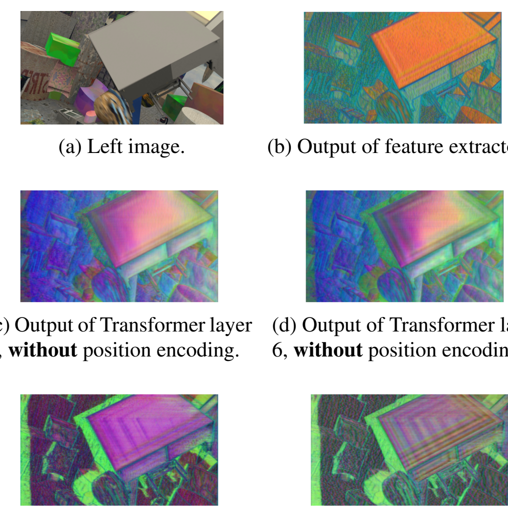
Uniqueness Constraint: The soft uniqueness constraint is imposed via the optimal transport layer by taking the interaction between pixels along the same epipolar line into

Table 1. Ablation study on Scene Flow dataset. AM: attention mask. OT: optimal transport. CAL: context adjustment layer. RPE: relative positional encoding.

Component				3 px		Occ
AM	OT	CAL	RPE	Error ↓	EPE ↓	IOU ↑
				3.61	0.92	0.78
✓				2.77	0.84	0.77
✓	✓			2.32	0.70	0.87
✓	✓	✓		2.21	0.63	0.88
✓	✓	✓	✓	**1.26**	**0.45**	**0.92**

account. We find it improves the result in all metrics, especially for occlusion IOU (row 3 in Table 1).

Relative Positional Encoding: To visualize the effect of positional encoding, we use PCA to reduce the feature map to $\mathbb{R}^3$. Given an image with a large textureless area, such as the table shown in Fig. 4(a), the features directly extracted from the feature extractor shown in Fig. 4(b) exhibit similar patterns. In the case of no positional encoding, as the layers progress deeper, the feature map merely changes throughout the process as shown in Fig. 4(c-d). By providing relative positional encoding to all layers, strides that are parallel to the edges emerge in layer 4 as shown in Fig. 4(e) and eventually the strides propagate to the entire region in layer 6 as shown in Fig. 4(f). This suggests that the Transformer needs relative positional information to resolve ambiguity in textureless areas. With positional encoding added to all layers, the result improves for all three metrics (row 5 in Table 1).



(a) Left image.
(b) Output of feature extractor.
(c) Output of Transformer layer 4, **without** position encoding.
(d) Output of Transformer layer 6, **without** position encoding.
(e) Output of Transformer layer 4, **with** position encoding.
(f) Output of Transformer layer 6, **with** position encoding.

Figure 4. Feature descriptor visualization. Full evolution of features updated by Transformer can be found in Appendix C.

Generalization of Attention: In principle, STTR allows the disparity range to scale with the image width since pixels are densely compared. Nonetheless, we evaluate if STTR indeed generalizes beyond the disparity range that it is trained on. We train another model by only computing losses on pixels with disparities smaller than 192 px ($0.2I_w$) and ignoring pixels outside this range. During testing, the maximal disparity prediction made within 1 px error margin is 458 px ($0.48I_w$), which demonstrates the generalization.

Attention Span: We analyze the attention span (i. e., the distribution of attention values over all pixels) of each layer of self- and cross-attention [33, 34]. Our results (detailed in Appendix D) show that both self- and cross-attention start from relatively global context (300 px, $0.31I_w$) and shift to the local context (114 px, $0.12I_w$ for self-attention and 15 px, $0.01I_w$ for cross-attention). Since the attention span decreases in deeper layers, we conclude that the global context is predominantly used in early layers but does not contribute substantially to disparity refinement in the late layers. This suggests an opportunity for future work to gradually reduce the search window based on previous layers' attention span to improve efficiency.

4.2. Comparison with Prior Work

Under multiple evaluation settings, we compare STTR with prior work spanning the primary learning-based stereo depth paradigms, including correlation-based AANet [39], 3D convolution-based PSMNet [5] and GANet-11 [43], correlation and 3D convolution hybrid approach GwcNet-g [16] and a classification-based Bi3D [2].

4.2.1 Scene Flow Benchmark Result

The pre-training results on Scene Flow are shown in Table 3. STTR performs on par with prior work when evaluated only on pixels with disparity less than 192 (2nd – 3rd columns). However, STTR outperforms prior work by a large margin in unconstrained settings due to its unbounded disparity estimation (4th – 5th columns) while the maximal disparity is fixed at $D = 192$ for the other methods. To compare fairly, we evaluate again with $D = 480$ (which covers all disparity values in the test dataset) for prior work. STTR's performance remains the *same* as $D = 192$ setting and comparable to prior work. Moreover, as shown qualitatively in Fig. 1 and quantitatively in Table 1, STTR can accurately identify occluded areas, which was not attempted in prior work.

4.2.2 Cross-Domain Generalization

We examine the domain generalization of STTR by comparing it to prior work also trained only on the Scene Flow synthetic dataset in Table 2. Note that we do not refine the models to the test dataset. For a fair comparison, we set maximum disparity $D = 192$ for prior work and only evaluate pixels in this range. We also trained prior work with the same asymmetric augmentation technique used for STTR to avoid inconsistency [38], while keeping the original training scheme (optimizer, learning rate, loss and num-

Table 2. Generalization *without* fine-tuning on MPI Sintel, KITTI 2015, Middlebury 2014, and SCARED dataset. **Bold** is best. ‡: models trained with asymmetric data augmentation. †: $s = 4$ for STTR due to memory constraint. OOM: out-of-memory. (W×H): image resolution.

	MPI Sintel † (1024 × 436)			KITTI 2015 (1242 × 375)			Middlebury 2014 (varies)			SCARED † (1080 × 1024)		
	3 px Error ↓	EPE ↓	Occ IOU ↑	3 px Error ↓	EPE ↓	Occ IOU ↑	3 px Error ↓	EPE ↓	Occ IOU ↑	3 px Error ↓	EPE ↓	Occ IOU ↑
PSMNet [5]	6.81	3.31	N/A	27.79	6.56	N/A	12.96	3.05	N/A	OOM	OOM	N/A
PSMNet ‡	7.93	3.70	N/A	7.43	1.39	N/A	10.24	2.02	N/A	OOM	OOM	N/A
GwcNet-g [16]	6.26	1.42	N/A	12.60	2.21	N/A	8.59	1.89	N/A	OOM	OOM	N/A
GwcNet-g ‡	5.83	**1.32**	N/A	6.75	1.59	N/A	6.60	1.95	N/A	OOM	OOM	N/A
AANet [39]	5.91	1.89	N/A	12.42	1.99	N/A	12.80	2.19	N/A	6.39	1.36	N/A
AANet ‡	6.29	2.24	N/A	7.06	**1.31**	N/A	9.57	**1.71**	N/A	3.99	**1.17**	N/A
STTR ‡	**5.75**	3.01	**0.86**	**6.74**	1.50	**0.98**	**6.19**	2.33	**0.95**	**3.69**	1.57	**0.96**

Table 3. Evaluation on Scene Flow. Model weights provided by authors. D: maximum disparity value. OOM: out of memory. N/A: model only infers within 192 range.

	D=192				D=480			
	disp <192		All pixels		disp <192		All pixels	
	3 px Error ↓	EPE ↓	3 px Error ↓	EPE ↓	3 px Error ↓	EPE ↓	3 px Error ↓	EPE ↓
PSMNet	2.87	0.95	3.31	1.25	3.09	0.92	3.60	1.03
GwcNet-g	1.57	0.48	2.09	0.89	1.60	0.50	1.72	0.53
AANet	1.86	0.49	2.38	1.96	N/A		N/A	
GANet-11	1.60	0.48	2.19	0.97	OOM		OOM	
Bi3D	1.70	0.54	2.21	1.16	OOM		OOM	
STTR	**1.13**	**0.42**	**1.26**	**0.45**	**1.13**	**0.42**	**1.26**	**0.45**

ber of epochs). Although asymmetric augmentation improves generalization in some cases (confirming findings of [38]), it is not consistent. Regardless, STTR generalizes comparably and maintains a high occlusion IOU across four datasets. STTR performs well in terms of 3 px Error, but not always for EPE. This is because if a pixel is wrongly identified as occlusion, a zero disparity is predicted, resulting in a large EPE. We visualize the generalization mechanism of STTR in Appendix E.

4.2.3 KITTI Benchmark Result

Since the KITTI benchmark provides a reasonable number of images for fine-tuning and it is commonly used in prior work, we choose the KITTI benchmark for comparisons after fine-tuning. The result on KITTI 2015 benchmark is shown in Table 4[2]. STTR performs comparably to several competing approaches, even compared with multi-resolution networks designed for better context aggregation.

4.2.4 Shortcomings in Challenge Design

While STTR performs comparatively to prior work, it is worth mentioning that the test sets in these real-world datasets are rather small. Specifically, the test sets accompanying KITTI 2015, Middlebury 2014, and SCARED contain 200, 15, and 19 images, respectively. To achieve a more comprehensive estimate of performance, much larger test datasets are required. This paucity of data is further compounded by the observation that performance differences between competing models on these datasets are on the order of $< 1\,\%$ after refinement and a few percent for cross-domain generalization. Consequently, we cannot conclude if there is a significant performance difference between the approaches presented. Additionally, KITTI only reports on disparities less than 192 and does not have metrics related to some of the benefits central to STTR (i. e., unlimited disparity and occlusion detection). As such, results on this benchmark likely cannot give a complete picture of performance comparisons.

Despite the above limitation, certain advancements have brought substantial performance improvements for cost-volume approaches. The shift from low-res (e. g. PSMNet [5]) to multi-res using feature pyramids for better context aggregation (such as LEAStereo [7]) appears to be a fruitful path, where LEAStereo [7] is further optimized within this structure using neural architecture search. Our future work aims to incorporate these developments into our design.

Table 4. Evaluation of 3 px or 5% Error on KITTI 2015. bg: background. fg: foreground. multi-res: networks operate on multiple resolutions. low-res: networks operate on downsampled resolution.

	Methods	Year	bg ↓	fg ↓	all ↓
multi-res	HSM-1.8 [40]	2019	1.63	3.40	1.92
	AMNet [12]	2019	1.39	3.20	1.69
	AANet [39]	2020	1.80	4.93	2.32
	LEAStereo [7]	2020	**1.29**	**2.65**	**1.51**
low-res	PSMNet [5]	2018	1.71	4.31	2.14
	GANet-15 [43]	2019	1.40	3.37	1.73
	GwcNet-g [16]	2019	1.61	3.49	1.92
	Bi3D [2]	2020	1.79	3.11	2.01
	STTR		1.70	3.61	2.01

5. Conclusion

In conclusion, we have presented an end-to-end network architecture named STereo TRansformer that synergizes advantages of CNN and Transformer architectures. We revisit stereo depth estimation from a sequence-to-sequence matching perspective. This approach 1) avoids the need to pre-specify a fixed disparity range, 2) explicitly handles occlusion, and 3) imposes a match uniqueness constraint. We experimentally demonstrate that STTR generalizes to different domains without fine-tuning and report promising results on benchmarks with refinement. Future work will include increasing the context information via multi-resolution techniques.

[2] Full result of KITTI benchmark.

References

[1] Max Allan, Jonathan Mcleod, Cong Cong Wang, Jean Claude Rosenthal, Ke Xue Fu, Trevor Zeffiro, Wenyao Xia, Zhu Zhanshi, Huoling Luo, Xiran Zhang, et al. Stereo correspondence and reconstruction of endoscopic data challenge. *arXiv preprint arXiv:2101.01133*, 2021.

[2] Abhishek Badki, Alejandro Troccoli, Kihwan Kim, Jan Kautz, Pradeep Sen, and Orazio Gallo. Bi3d: Stereo depth estimation via binary classifications. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1600–1608, 2020.

[3] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *European conference on computer vision*, pages 611–625. Springer, 2012.

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. *arXiv preprint arXiv:2005.12872*, 2020.

[5] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5410–5418, 2018.

[6] Zhuoyuan Chen, Xun Sun, Liang Wang, Yinan Yu, and Chang Huang. A deep visual correspondence embedding model for stereo matching costs. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 972–980, 2015.

[7] Xuelian Cheng, Yiran Zhong, Mehrtash Harandi, Yuchao Dai, Xiaojun Chang, Tom Drummond, Hongdong Li, and Zongyuan Ge. Hierarchical neural architecture search for deep stereo matching. *arXiv preprint arXiv:2010.13501*, 2020.

[8] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

[9] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in neural information processing systems*, pages 2292–2300, 2013.

[10] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*, 2019.

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[12] Xianzhi Du, Mostafa El-Khamy, and Jungwon Lee. Amnet: Deep atrous multiscale stereo disparity estimation networks. *arXiv preprint arXiv:1904.09099*, 2019.

[13] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[14] Andreas Griewank and Andrea Walther. Algorithm 799: revolve: an implementation of checkpointing for the reverse or adjoint mode of computational differentiation. *ACM Transactions on Mathematical Software (TOMS)*, 26(1):19–45, 2000.

[15] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2495–2504, 2020.

[16] Xiaoyang Guo, Kai Yang, Wukui Yang, Xiaogang Wang, and Hongsheng Li. Group-wise correlation stereo network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3273–3282, 2019.

[17] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*, 2020.

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

[19] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[20] 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 *Proceedings of the IEEE International Conference on Computer Vision*, pages 66–75, 2017.

[21] Hamid Laga, Laurent Valentin Jospin, Farid Boussaid, and Mohammed Bennamoun. A survey on deep learning techniques for stereo-based depth estimation. *arXiv preprint arXiv:2006.02535*, 2020.

[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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2811–2820, 2018.

[23] Xingtong Liu, Yiping Zheng, Benjamin Killeen, Masaru Ishii, Gregory D Hager, Russell H Taylor, and Mathias Unberath. Extremely dense point correspondences using a learned feature descriptor. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4847–4856, 2020.

[24] Yanbin Liu, Linchao Zhu, Makoto Yamada, and Yi Yang. Semantic correspondence as an optimal transport problem. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4463–4472, 2020.

[25] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4040–4048, 2016.

[26] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3061–3070, 2015.

[27] Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael

Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. Mixed precision training. *arXiv preprint arXiv:1710.03740*, 2017.

[28] Yuichi Ohta and Takeo Kanade. Stereo by intra-and inter-scanline search using dynamic programming. *IEEE Transactions on pattern analysis and machine intelligence*, (2):139–154, 1985.

[29] 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*, pages 8026–8037, 2019.

[30] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

[31] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4938–4947, 2020.

[32] 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 *German conference on pattern recognition*, pages 31–42. Springer, 2014.

[33] Sainbayar Sukhbaatar, Edouard Grave, Piotr Bojanowski, and Armand Joulin. Adaptive attention span in transformers. *arXiv preprint arXiv:1905.07799*, 2019.

[34] Yi Tay, Mostafa Dehghani, Samira Abnar, Yikang Shen, Dara Bahri, Philip Pham, Jinfeng Rao, Liu Yang, Sebastian Ruder, and Donald Metzler. Long range arena: A benchmark for efficient transformers. *arXiv preprint arXiv:2011.04006*, 2020.

[35] Stepan Tulyakov, Anton Ivanov, and Francois Fleuret. Practical deep stereo (pds): Toward applications-friendly deep stereo matching. *arXiv preprint arXiv:1806.01677*, 2018.

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[37] Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. *arXiv preprint arXiv:2003.07853*, 2020.

[38] Jamie Watson, Oisin Mac Aodha, Daniyar Turmukhambetov, Gabriel J Brostow, and Michael Firman. Learning stereo from single images. In *European Conference on Computer Vision*, pages 722–740. Springer, 2020.

[39] Haofei Xu and Juyong Zhang. Aanet: Adaptive aggregation network for efficient stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1959–1968, 2020.

[40] Gengshan Yang, Joshua Manela, Michael Happold, and Deva Ramanan. Hierarchical deep stereo matching on high-resolution images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5515–5524, 2019.

[41] Jiayu Yang, Wei Mao, Jose M Alvarez, and Miaomiao Liu. Cost volume pyramid based depth inference for multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4877–4886, 2020.

[42] Jiahui Yu, Yuchen Fan, Jianchao Yang, Ning Xu, Zhaowen Wang, Xinchao Wang, and Thomas Huang. Wide activation for efficient and accurate image super-resolution. *arXiv preprint arXiv:1808.08718*, 2018.

[43] Feihu Zhang, Victor Prisacariu, Ruigang Yang, and Philip HS Torr. Ga-net: Guided aggregation net for end-to-end stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 185–194, 2019.