Parallax attention stereo matching network based on the improved group-wise correlation stereo network

Xuefei Yu, Jinan Gu*, Zedong Huang, Zhijie Zhang

School of Mechanical Engineering, Jiangsu University, Zhenjiang 212000, China

* gujinan@tsinghua.org.cn

Abstract

Recent stereo matching methods, especially end-to-end deep stereo matching networks, have achieved remarkable performance in the fields of autonomous driving and depth sensing. However, state-of-the-art stereo algorithms, even with the deep neural network framework, still have difficulties at finding correct correspondences in near-range regions and object edge cues. To reinforce the precision of disparity prediction, in the present study, we propose a parallax attention stereo matching algorithm based on the improved group-wise correlation stereo network to learn the disparity content from a stereo correspondence, and it supports end-to-end predictions of both disparity map and edge map. Particular, we advocate for a parallax attention module in three-dimensional (disparity, height and width) level, which structure ensures high-precision estimation by improving feature expression in near-range regions. This is critical for computer vision tasks and can be utilized in several existing models to enhance their performance. Moreover, in order to making full use of the edge information learned by two-dimensional feature extraction network, we propose a novel edge detection branch and multi-featured integration cost volume. It is demonstrated that based on our model, edge detection project is conducive to improve the accuracy of disparity estimation. Our method achieves better results than previous works on both Scene Flow and KITTI datasets.

Introduction

The binocular stereo matching task is an imperative, but difficult scientific problem, which aims at computing disparity data for every pixel from a stereo correspondence. Efficient and correct stereo matching methods are necessary for computer vision tasks such as robotic pose estimation and autonomous driving [1,2].

Traditional stereo matching methods usually consist of four steps: initial matching cost calculation, matching cost aggregation, disparity prediction, and post-processing. These can be categorized into global and local algorithms [3]. Local strategies solely use constant measurement windows or changeable windows to calculate the preliminary cost. Global strategies normally treat an optimization task by minimizing a word goal characteristic that incorporates
However, traditional algorithms need to manually design feature description operators and cost aggregation strategies, which is not suitable for real-time applications. Complicated hand-craft production steps limit their improvement.

Learning-based stereo-matching methods achieve accurate matching of the corresponding points in the left and right feature maps through exploring feature representations and aggregation algorithms. Those algorithms commonly consist of the following four steps: unary characteristic extraction [4], constructing cost volume [5], value aggregation [6], and disparity prediction [7]. Although the performance on several benchmarks is significantly promoted, drawbacks still remain: Firstly, the predicted edge cues of the disparity map is not accurate enough. Secondly, adopting the strategy of global attention, which is insensitive to the detailed texture regions, resulting in inaccurate disparity estimation in vital areas.

In recent years, researchers are encouraged by the mechanism of human attention and attempt to design some network attention architectures with a CNN to enhance the performance of feature extraction. However, drawbacks still remain: Firstly, these works focus on parallax information in two-dimensional (2D) domain, 2D feature is difficult to fully reflect the three-dimensional (3D) real scenes, while ignoring the more important 3D information. Secondly, due to limited learning capability of a single network structure, the disparity map predicted is not fine enough in near-range regions.

The edge cues of images are the most easily recognized feature by human eyes, in other words, humans can easily find the stereo correspondence by using edge cues of binocular images. Based on this observation, some researches have made partial progress in predicting image edge cues as a single task. In recent years, researchers are encouraged by edge detection (ED) task and use it for disparity estimation project. However, these methods regard disparity prediction and edge detection as a multi-task learning project. Yet, features learned in such multi-task pipelines cannot be fully exploited, which poses a great need for an effective fusion mechanism.

In the context of autonomous driving, the relatively closer area provides larger parallax information, leading to greater risks. To address this problem, more attention should be assigned to this kind of region in the disparity estimation model. In this paper, we propose a high-quality and efficient module for stereo matching and our method achieves better performance on SceneFlow [8] and KITTI [9,10] than previous methods. Specifically, we examine the very important issue of structure design, attention. The importance of attention has been researched particularly in previous methods [11,12]. In our module, the left characteristic map \( f_l \) and the corresponding right feature map \( f_r \) are packed in the shape of a 3D feature map \( f \), which is sent to the parallax attention (PA) stereo module to learn ‘what is’ to attend \( f \). As shown in Fig 1, our structure efficiently improves the accuracy of disparity prediction by improving feature expression in near-range regions. Meanwhile, a novel edge detection branch and a multi-featured integration cost volume are proposed in our network to learn finer texture features, which are vital in the optimization of unary feature extraction tasks. In order to complete the end-to-end disparity prediction task, we assign different weights to edge detection loss and disparity smooth loss. It is demonstrated that achieving high-precision edge feature map is conducive to improve the accuracy of disparity estimation.

Our main contributions can be summarized as follows:

1. We propose a PA module to further improve the accuracy of disparity prediction;
2. An edge detection branch and a multi-featured integration cost volume are proposed in our network architecture to obtain finer texture features;
3. Our PA-Net achieves the accuracy of 0.775 end-point-error (EPE) on Scene flow dataset and 2.05% kitti-d1-all error on KITTI 2015 dataset, which outperforms other methods by 12%.

Related work

2.1. Traditional methods

In non-end-to-end depth stereo matching algorithms, each step of traditional stereo matching can be replaced by a neural network. Some researchers have mainly focused on the use of CNN to accurately calculate the matching cost function and use the semi-global matching [14,15] method to optimize the predicted disparity map. Zbontar et al. [16] proposed a network structure called stereo matching by training a convolutional neural network (MC-CNN) to compare image patches to calculate the cost of matching by utilizing a pair of 9x9 patches. Traditional algorithms play an important role in stereo matching tasks. However, traditional algorithms generally face the problems of slow calculation speed and low matching accuracy, which greatly limits the application of stereo matching algorithm.

2.2. Learning-based methods

In 2015, Long et al. [17] achieved very good results in semantic segmentation using a fully convolutional network (FCN). Mayer et al. [8], inspired by the FCN, introduced an end-to-end stereo network in an optical flow prediction task. Disp-Net calculates the Euclidean distance construction loss for each pixel between the estimated disparity map and real disparity value. Cascade residual learning: A two-stage convolutional neural network for stereo matching (CRL) [18], and learning for disparity estimation through feature constancy. (iRes-Net) [19], utilized the idea of DispNetC [8] with stack refinement structures to optimize stereo results. Kendal et al. [20] proposed an end-to-end network GC-Net, which considers the use of context and scene geometry information in stereo matching. GC-Net is the first to concatenate the left \( f_l \) and the right feature \( f_r \) to form a 4D cost volume:

\[
C_{\text{concatenation}}(x, y, d) = \text{Concat}\{f_l(x, y), f_r(x - d, y)\}. \tag{1}
\]

Meanwhile, GC-Net transforms the stereo matching problem into a regression problem and directly realizes a refined output without post-processing. Encouraged by GC-Net, Chang et al. [21] proposed a PSM-Net, combining the spatial pyramid pooling and stacked 3D
hourglass structures in a stereo refinement network. In the current work, GWC-Net [13] proposes a group-wise correlation to assemble the cost volume, whose idea is splitting the features into groups and computing correlation maps group by group. The group-wise correlation is computed as

\[
C_{\text{group-wise}}(d, x, y, g) = \frac{1}{N_c/N_g} \langle f^g(x, y). f^g(x - d, y) \rangle,
\]

where \(N_c\) denotes the channels of unary features and it eventually divided into \(N_g\) groups along channel dimension. \(\langle \cdot, \cdot \rangle\) is the internal product at all disparity levels \(d\). Although the performance on several benchmarks is significantly promoted, there remains some drawbacks, including the predicted edge contour of the disparity map is not accurate enough and adopting the strategy of global attention, which is insensitive to the detailed texture information.

2.3. Learning-based attention methods

In recent years, researchers are encouraged by the mechanism of human attention and attempt to design some network attention architectures with a CNN to enhance the performance of feature extraction. Hu et al. [22] introduced a squeeze-and-excitation block to fully utilize the channel information in the network. In addition to channel attention, cbam: convolutional block attention module [23] introduced a spatial attention block to demonstrate that spatial features are vital in the network. Wang et al. [24] introduced a PASSR-Net to integrate super-resolution information from a stereo image pair, and proposed a PAM module in the article of PASMnet [25] to calculate the consistency score of left and corresponding right graphs along the epipolar line, and it was leveraged by many subsequent methods such as [19–26]. On the basis of PAM, Wang et al. [27] introduced a symmetric bi-directional parallax attention module (biPAM) to obtain cross-view information. Ying et al. [28] proposed a generic stereo attention module (SAM) which aims to solve the information incorporation problem. Chen et al. [29] addressed the stereo images with large disparity and multi types of epipolar lines issues by utilizing a cross parallax attention module (CPAM). However, the parallax information provided by binocular images has not been fully utilized in those methods. PA-Net is the first to emphasize that by improving feature expression in near-range regions is helpful to disparity prediction task.

2.4. Edge detection methods

Edge cues can be easily captured by human eyes to find stereo correspondences. Accurate edge contours can help discriminating between different objects or regions. Based on this observation, some works had made some progress in predicting image edge cues as a single task. Xie et al. [30,31] first designed an end-to-end ED network based on a VGG-16 network. Recently, Song et al. [32,33] combined an ED branch with stereo-matching network. However, these methods regard disparity prediction and edge detection as a multi-task learning project, those works did not establish an effective mechanism to integrate the information learned by multi-task project. As a result, the features learned by multi-task project are not effectively expressed and utilized. Focus on this problem, we construct a multi-featured integration cost volume to combine parallax features and edge features.

Methods

As shown in Fig 2, we proposed a PA stereo matching network (PA-Net), which extends GWC-Net [13] with a PA module, edge detection branch, and multi-featured integration cost volume.
3.1. Network architecture

The pipeline of our introduced PA network is shown in the upper half of Fig 2, it includes four parts: unary feature extraction pipeline, multi-featured integration cost volume structure, parallax-attention 3D aggregation network, and disparity prediction module. The multi-featured integration cost volume structure consists of three parts: concatenation [20], group-wise correlation [13], and edge detection volumes (details in Section 3.3). The results of the multi-featured integration cost volume are then concatenated as the input of the parallax-attention 3D aggregation network, and it will be described in Section 3.2.

The parallax-attention 3D aggregation network aims to aggregate variable disparity values, which consist of two parts: a pre-hourglass module and three parallax-attention 3D aggregation networks. The pre-hourglass module consists of two components: the primary half consists of four 3D convolutional layers with batch normalization and the ReLU [26] function, where the second part consists of two PA modules.

3.2. Parallax attention module

Discriminant characteristic representations are essential for understanding the scenes. However, previous studies only focus on the two-dimensional (2D) contextual information, but ignore the significance of 3D disparity features. To emphasize the value of regions with a large parallax, we introduce a PA module that encodes the disparity information to different weights, thus enhancing their illustration capability.

3D convolution layer is widely used in stereo matching tasks and it consists of 4 parts: channel dimension, disparity dimension, height and width. However, 3D filters learned within a local field that lacks contextual information in the output feature map $U$.

Based on these observations, as is shown in Fig 3, given a feature map $f \in R^{C \times D \times H \times W}$, we first conduct two transformations $f \rightarrow f_{ij} \in R^{C \times H \times W}$, $f \rightarrow f_{m} \in R^{C \times H \times W}$, which represent the
importance of different position vectors. Term \( f_s \) implies to select the maximum value element in disparity dimension, whereas \( f_m \) denotes to calculate the mean value in disparity dimension, with regard to the \( c^{th} \) channel, the value of \((i, j)\) position is calculated by:

\[
f_m(c; i, j) = \frac{1}{d_{\text{max}}} \sum_{d=1}^{d_{\text{max}}} U_c(i^d, j^d),
\]

where \( d_{\text{max}} \) denotes the maximum parallax value. Thereafter, the two feature maps are concatenated in the channel dimension to obtain a mixed feature map \( M_u \in \mathbb{R}^{RC \times 1 \times H \times W} \). The mixed feature map \( M_u \) can be treated as a collector of the local disparity texture information, and its function is to describe the entire parallax image.

Subsequently, the feature map is sent to a shared network, which is composed of a multi-layer perceptron with two \( 3 \times 3 \times 3 \) convolutional layers and it accompanies the batch normalization and ReLU [26] function. To reduce the parameter overhead, the characteristics of the middle layer are set to \( \mathbb{R}^{CR \times 1 \times H \times W} \), where \( r \) denotes the reduction ratio, and we set it to 4. Additionally, the disparity feature map is applied to a sigmoid function. Finally, we merge the output feature vectors with the input feature \( f \) using an element-wise product to obtain the final PA feature map \( M_d \in \mathbb{R}^{C \times D \times H \times W} \), which can be simplified as follows:

\[
M_{dx_i} = \frac{X_i}{1 + \exp^{-(\text{MLP}(\langle X_i \rangle f_m(X_i)))}},
\]

where \( M_{dx_i} \) denotes the value of the final \( i^{th} \) position, \( \langle \cdot, \cdot \rangle \) means concatenating the inner channels, and \( X_i \) denotes the value of the input feature.

In comparison with traditional 3D convolutional layers, our contributions can be summarized as follows:

1. In the case of acquiring an identical receptive field, our module generates considerably fewer parameters (reduced by 25%) and consumes much less memory; consequently, the inference time of our module is faster.
2. As summarized in Table 1, our PA structure can effectively decrease the performance of EPE with a small increase of computational complexity.
3. Our PA module does not change the number of channels and the size of input features, which can be added directly to 3D convolution layers.
3.3. Edge detection and multi-featured cost volume

State-of-the-art disparity estimation method works well on ordinary and clear texture regions. The matching clues in these regions are clear and can be easily captured through the context pyramid. However, as shown in Fig 1, the edge details are lost. Hence, we design an edge detection branch to help modify disparity map.

Our edge detection (ED) architecture includes three branches (group-wise, concatenation, and edge detection branches), sharing the same weights of the ResNet-50 backbone, listed in Table 1. There are four outputs in the ResNet-50 layer, for each output branch, we design a new structure that includes a $3 \times 3$ convolutional layer and $1 \times 1$ convolutional layer with batch normalization and the ReLU [26] function (we set the number of the final channel to 1 in each branch); In order to fuse the features contained in different branches, all the feature maps are concatenated to construct an edge cost volume. Finally, group-wise, concatenation, and ED features are fused to form a multi-featured integration cost volume.

As shown in Fig 4, based on the architecture of the GWC-Net [13], we added an ED branch to construct the edge volume. In contrast to the concatenation volume, within which the left and corresponding right feature maps are concatenated at unique disparity levels to form a cost volume, the ED volume is constructed by computing the similarities of the left and right

![Diagram of feature extraction network](https://doi.org/10.1371/journal.pone.0263735.g004)

Table 1. Ablation study results of PSM-Net, GWC-Net and PA-Net on the datasets of Scene flow [8]. The results of PSM-Net [21] and GWC-Net [13] are trained with published code with our batch size, evaluation settings for fair comparison.

| Model       | Edge detection | PA module | EPE (px) | >1px (%) | >2px (%) | >3px (%) | Time (ms) |
|-------------|----------------|-----------|----------|----------|----------|----------|-----------|
| PSM-Net     | ✓              | ✓         | 0.988    | 10.62    | 6.31     | 5.02     | 246.1     |
|             | ✓              | ✓         | 0.955    | 10.27    | 6.10     | 4.85     | 251.5     |
| GWC-Net     | ✓              | ✓         | 0.892    | 10.16    | 6.31     | 4.80     | 259.4     |
|             | ✓              | ✓         | 0.878    | 9.25     | 5.57     | 4.35     | 210.7     |
|             | ✓              | ✓         | 0.856    | 9.08     | 5.47     | 4.27     | 215.6     |
| PA-Net(ours)| ✓              | ✓         | 0.792    | 8.59     | 5.24     | 4.08     | 224.4     |

https://doi.org/10.1371/journal.pone.0263735.t001
feature maps. For each pair of edge feature maps, the edge correlation is calculated as follows:

\[ C_{\text{edge}}(d, x_i, y_j) = \frac{1}{N_c} \sum_{i,j>0} \langle f_{\text{edge}}(x_i, y_j), f_{\text{edge}}(x_i - d, y_j) \rangle, \]  

(5)

where \( \langle \cdot, \cdot \rangle \) denotes the internal product, \( N_c \) the quantity of total channels, and \( f_{\text{edge}}, \bar{f}_{\text{edge}} \) the left and corresponding right feature maps, respectively. Finally, by combining all cost volumes, the multi-scale cost volume is determined as follows:

\[ C_{\text{Multi-scale}} = \text{Concat}(C_{\text{edge}}, C_{\text{concatenation}}, C_{\text{group-wise}}), \]  

(6)

where \( \text{Concat}(\cdot, \cdot) \) denotes the concatenation of feature maps in the channel dimension. \( C_{\text{concatenation}} \) and \( C_{\text{group-wise}} \) are calculated as introduced in the Eqs of (1) and (2).

### 3.4. Output module and loss function

Summarizing all outputs of our network, it includes 4 disparity predicted maps \( d_s \), to fully utilize the output feature maps, we assign different weights to each output. We first employ two convolutional layers with \( 3 \times 3 \) and \( 1 \times 1 \) to obtain a 1-channel output; thereafter, the output feature map is upsampled using bilinear interpolation. Finally, a softmax function is designed to calculate the disparity prediction map. Generally, the disparity smooth loss can be calculated as follows:

\[ L_d(d_s, \tilde{d}_s) = \sum_{i=1}^{4} \lambda_i \left[ \frac{1}{N} \sum_{j=1}^{N} \text{smooth}_{L_1}(d^i_j - \tilde{d}^i_j) \right], \]  

(7)

where \( \lambda_i \) denotes the weight for the \( i^{th} \) output disparity prediction map, \( N \) represents the total number of pixels in one image, and \( d^i_j \) is the \( j^{th} \) element with ground truth \( \tilde{d}^i_j \). The \( \text{smooth}_{L_1} \) loss is computed as follows:

\[ \text{smooth}_{L_1}(x) = \left\{ \begin{array}{ll}
\frac{x^2}{2}, & \text{if } |x| < 1 \\
|x| - 0.5, & \text{otherwise}
\end{array} \right. \]  

(8)

Since the information of object edge contour in images is conducive to the parallax prediction task, we propose an edge detection loss for end-to-end learning:

\[ L_e(x) = \frac{1}{N} \sum_{j=1}^{N} \left\{ \log(1 - P(x^j)), \text{ if } y^j = 0 \\
\log P(x^j), \text{ otherwise} \right. \]  

(9)

where \( x^j \) and \( y^j \) represent the activation value and ground truth edge probability at pixel \( j \), respectively. \( P(x) \) is the standard sigmoid function, and \( N \) denotes the total number of pixels in one image. Fusing the edge feature information extracted from different output layers, our edge loss function can be formulated as:

\[ L_e = \sum_{k=1}^{3} \beta_k L_e(x^k) + L_e(x^\text{fuse}), \]  

(10)

where \( x^k \) is the activation value from stage \( k \) while \( x^\text{fuse} \) denotes the last edge output. \( \beta_k \) is the weight of stage \( k \) (equals to 0.2, 0.4, and 0.6 here). Since we are working under a disparity prediction task setup, we want to fuse the edge detection loss and disparity prediction loss together. Therefore, we design a double hierarchical loss weighing scheme, the total loss is
calculated as:

\[ L = \gamma_0 L_d + \gamma_1 L_e, \quad (11) \]

with \( \gamma_0 \) is the weight of total disparity prediction loss and it set to 1, \( \gamma_1 \) denotes the total edge detection loss weighted 0.1.

**Experiments**

In this section, we evaluated our PA-Net with distinctive settings on the Scene Flow [8] and the KITTI datasets [9,10]. Sections 4.1 and 4.2 show the experimental setup of proposed network on the KITTI and Scene Flow datasets. In Section 4.3, we set up a series of ablation experiments using different methods to test the performance of our PA module. In Section 4.4, we add our edge detection volume to PSM-Net and GWC-Net to validate the importance of the multi-featured integration cost volume.

**4.1. Experimental setup**

We implemented our architectures using the PyTorch tools. All methods were trained using Adam (\( \beta_1 = 0.9, \beta_2 = 0.99 \)). Owing to the limitation of experimental conditions, the batch size of our network was set to 4, and we optimized all the models with two Nvidia RTX 2080ti GPUs using 256×512 random crops from the input image pair. The utmost disparity value was set to 192, whereas the coefficients of the four outputs were set to \( \lambda_1 = 0.5, \lambda_2 = 0.5, \lambda_3 = 0.7, \) and \( \lambda_4 = 1.0 \). We tend to set the model on the Scene Flow dataset for a total of 16 epochs in which the learning rate was 0.001 and downscaled by 2 when the number of epochs 10, 12, and 14. For KITTI [9,10] dataset, the pre-trained model on Scene Flow [8] datasets was utilized to train another 300 epochs. The preliminary learning rate was 0.001, it is down-scaled by 10 when exceeding 200 epochs.

**4.2. Dataset**

1) Scene Flow [8]: A dataset series of artificial stereo datasets. There are three subsets in the dataset: Driving, Flyingthings3D, and Monkaa, containing 35454 images for training and 4370 images for testing with Height = 540 and Width = 960, along with ground truth maps. The trained network model has a strong generalization ability because the number of pictures in the Scene Flow dataset is sufficiently large. The results of visualization and comparisons for Scene Flow [8] are as shown in Fig 1.

2) KITTI 2012 and KITTI 2015 [9,10]: Real-word driving scene dataset using Lidar scanning to obtain the three-dimensional coordinates of space points. KITTI 2012 includes 194 training stereo correspondences and 195 testing pairs. KITTI 2015 comprises 200 stereo correspondences for training and testing. The training dataset is divided into two parts, the first section consists of 180 pairs for training and the relaxation groups of images for validation. More than that, we made the corresponding edge detection label dataset for end-to-end edge detection task. The results of the visualization and comparisons are shown in Fig 5, and we submit the results predicted by PA model on the validation set of the KITTI official website. The comparison results for the test set are summarized in Tables 2 and 3. It shows that our PA-Net achieves better results than PSMNet [21], GwcNet [13] and PASMNet [25].

**4.3 Ablation study for parallax attention module**

In this section, to validate the performance of the PA module, we evaluated the PA module with different stereo matching strategies. Moreover, we set a series of ablation experiments to
explore the best settings for the number of PA modules. [13,21] were selected as reference models by adding PA modules.

PA module can be directly used in 3D convolution layer, since our model will not change the number of channels and image size. The experimental results demonstrate that, on the premise of a small increase in calculation time, our PA-Net performs better than previous works. As summarized in Table 1, we select the classic methods [13,21] as the conference models which include two variables (edge detection structure and PA module). Meanwhile, the prediction result of an EPE on the Scene Flow dataset is decreased by 9.71% in the model of [21] and decreased by 11.0% in the model of [13] after adding the PA modules. Fig 6, depicts the training and validation curves of PA-Net, GWC-Net and PSM-Net on KITTI 2015 dataset from epoch 100 to 300. We can easily observe that the loss curve of PA-Net decreases more smoothly than previous works and produce consistent gains in performance which are sustained throughout the training process. Moreover, we see that PA-Net performs better than those works when the networks are trained to start fitting, and it will last achieve a highest accuracy.

To select an optimal value of PA modules to configure the networks. As shown in Table 4, which indicates the consequence of PA-Net with different numbers of PA modules. When the number of PA modules is larger than 6, the increase in accuracy becomes minor. Considering the amount of calculation and memory consumption, we selected half a dozen PA modules as the ultimate structure.

---

**Table 2. Performance comparison of KITTI 2012 [9] test set.** The GWC-Net [13] and PSM-Net [21] are trained with the same batch size as our method for fair comparison.

| Methods       | >2px (%) | >3px (%) | >5px (%) | Mean Error (px) | Time (s) |
|---------------|----------|----------|----------|-----------------|----------|
|               | Noc      | All      | Noc      | All             |          |
| DispNetC [8]  | 7.38     | 8.11     | 4.11     | 4.65            | 2.05     | 2.39     | 0.9      | 1.0      | 0.06     |
| MC-CNN-act [16]| 3.90   | 5.45     | 2.43     | 3.63            | 1.64     | 2.39     | 0.7      | 0.9      | 0.67     |
| G-Net [20]    | 2.71     | 3.46     | 1.77     | 2.30            | 1.12     | 1.46     | 0.6      | 0.7      | 0.9      |
| iResNet [19]  | 2.69     | 3.34     | 1.71     | 2.16            | 1.06     | 1.32     | 0.5      | 0.6      | 0.12     |
| PSMNet [21]   | 2.68     | 3.20     | 1.68     | 2.09            | 1.05     | 1.21     | 0.5      | 0.5      | 0.6      |
| GWCNet [13]   | 2.21     | 2.88     | 1.40     | 1.81            | 0.85     | 1.11     | 0.5      | 0.5      | 0.32     |
| Edge-stereo [32]| 2.32   | 2.88     | 1.46     | 1.83            | 0.83     | 1.04     | 0.4      | 0.5      | 0.32     |
| PA-Net        | **2.20** | **2.82** | **1.36** | **1.78**        | **0.80** | 1.09     | **0.5**  | **0.5**  | **0.33** |
4.4 Ablation study for multi-featured cost volume

In this section, we apply several critical modifications to the feature extraction network compared to [13]. Specifically, we design a multi-featured integration cost volume structure that consists of three parts: ED cost, group-wise cost, and concatenation cost volumes.

The experimental results in Table 1 demonstrate that by adding our edge detection structure, the parameters of EPE loss can be reduced appropriately. As summarized in Table 1, the prediction results of EPE on the Scene Flow dataset are decreased by 3.34% in the model of [21] and decreased by 2.50% in the model of [13] after adding the ED modules.

Based on [13], we can conclude from several experiments that if we set the channel number of the group-wise volume as 32, we can obtain an exceptional performance. The experimental consequences in Table 5 demonstrate that the EPE is decreased by adding the correct channels of the concat volume. The best EPE is 0.574 (concat channels are $14 \times 2$) in the dataset of KITTI 2015 and 0.616 (concat channels are $16 \times 2$) in KITTI 2012. Considering both the accuracy of disparity prediction and computer memory usage, we selected $14 \times 2$ as the ultimate channel of the concatenation volume.

4.5 Analysis and interpretation

While PA blocks have been empirically shown to improve network performance, we would like to provide an explain how the parallax attention mechanism operates in practice. To provide a clearer picture of the behavior of PA blocks, in this section we apply several examples from GWC-Net model and examine the different distributions of sensitive respective region between 3D convolutional layers and PA blocks. We then exhibit their distribution maps in Fig 7, which is trained in the dataset of KITTI 2015.
We make the following observations about how the parallax attention mechanism works in 3D feature extraction stage. First, traditional 3D convolutional layer is used to adopting global receptive field mechanism, which will guide the network to pay fair attention to different features. However, in the practice life, we can easily draw a conclusion that objects closer to us will have a greater impact. As shown in Fig 7, compared with trees and houses far away, people and cars in near region should be paid more attention. But we can observe a phenomenon that the lighted regions in 3D feature map have covered every corner of the image, which is not in line with objective reality. Second, for PA blocks, we redistribute the values in the parallax dimension to make full use of context information. For the region with large parallax value, the proportion of its value will be larger after redistribution. In the third line of Fig 7, lighted regions are concentrated in areas with large parallax such as roads, cyclist and cars, indicating that the network pays more attention to these areas. PA blocks successfully focus on objects with large parallax through the weight redistribution strategy in parallax dimension.

**Discussion**

Intuitively, our method not only improves the accuracy of disparity prediction globally, we also achieve the following advantages: Firstly, in the case of acquiring an identical receptive field with traditional 3D convolutional layer, our module generates considerably fewer parameters (reduced by 25%) and consumes much less memory. More than that, PA module can be easily utilized in other works because it will not change the size of the feature image. Secondly, as shown in Fig 1, our structure efficiently improves the accuracy of disparity prediction in near-range regions by improving 3D feature expression. Lastly, in order to making full use of the edge information learned by two-dimensional feature extraction network, we propose a novel edge detection branch and multi-featured integration cost volume. It is demonstrated that based on our model, edge detection project is conducive to improve the accuracy of disparity estimation. As Table 2 shows, compared with GWC-Net, our method performs better in two-pixel error, three-pixel error, and five-pixel error on the KITTI 2012 dataset. Compared with PSM-Net, the disparity map’s percentage of outliers averaged over all ground truth pixels (D1-all) is reduced by 11.6%, and the running speed is increased by 19.5%.

| Model       | The numbers of PA modules | Scene Flow EPE (px) | Time (s) |
|-------------|---------------------------|---------------------|----------|
| PA-Net      | 0                         | 0.856               | 0.329    |
|             | 1                         | 0.849               | 0.331    |
|             | 4                         | 0.816               | 0.334    |
|             | 6                         | 0.784               | 0.336    |
|             | 10                        | 0.790               | 0.340    |
|             | 16                        | 0.801               | 0.346    |

https://doi.org/10.1371/journal.pone.0263735.t004

| Model | Concatenation channels | Kitti2015 EPE (%) | Kitti2012 EPE (%) | Time (ms) |
|-------|------------------------|------------------|------------------|-----------|
| PA-Net (edge = 4, group = 32) | 10×2 | 0.612 | 0.633 | 215.7 |
|       | 12×2 | 0.586 | 0.621 | 218.5 |
|       | 14×2 | 0.574 | 0.617 | 222.4 |
|       | 16×2 | 0.578 | 0.616 | 225.9 |
|       | 18×2 | 0.575 | 0.619 | 229.4 |

https://doi.org/10.1371/journal.pone.0263735.t005
Conclusions

In this paper, we proposed a high-precision and practical stereo matching network, PA-Net, for end-to-end disparity prediction. Our net emphasizes that by improving feature expression in near-range regions is helpful to disparity prediction task. PA-Net performs better than previous networks by utilizing the edge detection layer, PA module, and multi-featured cost volume. It is demonstrated that based on our model, edge detection task is conducive to improve the accuracy of disparity estimation task. PA-Net achieves better accuracy than GWC-Net on the Scene Flow and KITTI datasets.

Acknowledgments

We would like to thank Editage (www.editage.com) for English language editing.

Author Contributions

Conceptualization: Xuefei Yu.
Data curation: Xuefei Yu.
Formal analysis: Xuefei Yu.
Funding acquisition: Xuefei Yu.
Investigation: Xuefei Yu.
Methodology: Xuefei Yu.
Project administration: Xuefei Yu.
Resources: Xuefei Yu, Zedong Huang.
Software: Xuefei Yu.
Supervision: Xuefei Yu, Jinan Gu, Zedong Huang, Zhijie Zhang.
Validation: Xuefei Yu.
Visualization: Xuefei Yu.
Writing – original draft: Xuefei Yu.
Writing – review & editing: Xuefei Yu.

References

1. Kerl C, Sturm J, Cremers D, editors. Robust odometry estimation for RGB-D cameras. 2013 IEEE international conference on robotics and automation; 2013: IEEE.
2. Shotton J, Girshick R, Fitzgibbon A, Sharp T, Cook M, Finocchio M, et al. Efficient human pose estimation from single depth images. 2012; 35(12):2821–40.
3. Scharstein D, Szeliski R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. 2002; 47(1):7–42.
4. Luo W, Schwing AG, Urtasun R, editors. Efficient deep learning for stereo matching. Proceedings of the IEEE conference on computer vision and pattern recognition; 2016.
5. Zbontar J, LeCun Y, editors. Computing the stereo matching cost with a convolutional neural network. Proceedings of the IEEE conference on computer vision and pattern recognition; 2015.
6. Zhang F, Prisacariu V, Yang R, Torr PH, editors. Ga-net: Guided aggregation net for end-to-end stereo matching. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2019.
7. Gidaris S, Komodakis N, editors. Detect, replace, refine: Deep structured prediction for pixel wise labeling. Proceedings of the IEEE conference on computer vision and pattern recognition; 2017.
8. Mayer N, Ilg E, Hausser P, Fischer P, Cremers D, Dosovitskiy A, et al., editors. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. Proceedings of the IEEE conference on computer vision and pattern recognition; 2016.
9. Geiger A, Lenz P, Urtasun R, editors. Are we ready for autonomous driving? the kitti vision benchmark suite. 2012 IEEE conference on computer vision and pattern recognition; 2012: IEEE.
10. Menze M, Geiger A, editors. Object scene flow for autonomous vehicles. Proceedings of the IEEE conference on computer vision and pattern recognition; 2015.
11. Mnih V, Heess N, Graves A, editors. Recurrent models of visual attention. Advances in neural information processing systems; 2014.
12. Xu K, Ba J, Kiros R, Cho K, Courville A, Salakhudinov R, et al., editors. Show, attend and tell: Neural image caption generation with visual attention. International conference on machine learning; 2015: PMLR.
13. Guo X, Yang K, Yang W, Wang X, Li H, editors. Group-wise correlation stereo network. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2019.
14. Hirschmuller H, editor Accurate and efficient stereo processing by semi-global matching and mutual information. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05); 2005: IEEE.
15. Hirschmuller H. Stereo Processing by Semi-GLOBAL Matching and Mutual Information, TPAMI Pattern Analysis and Machine Intelligence. IEEE; 2008.
16. Zbontar J, LeCun YJMLR. Stereo matching by training a convolutional neural network to compare image patches. 2016; 17(1):2287–318.
17. Long J, Shelhamer E, Darrell T, editors. Fully convolutional networks for semantic segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition; 2015.
18. Pang J, Sun W, Ren JS, Yang C, Yan Q, editors. Cascade residual learning: A two-stage convolutional neural network for stereo matching. Proceedings of the IEEE International Conference on Computer Vision Workshops; 2017.
19. Liang Z, Feng Y, Guo Y, Liu H, Chen W, Qiao L, et al., editors. Learning for disparity estimation through feature constancy. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2018.
20. Kendall A, Martirosyan H, Dasgupta S, Henry P, Kennedy R, Bachrach A, et al., editors. End-to-end learning of geometry and context for deep stereo regression. Proceedings of the IEEE International Conference on Computer Vision; 2017.
21. Chang J-R, Chen Y-S, editors. Pyramid stereo matching network. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2018.

22. Hu J, Shen L, Sun G, editors. Squeeze-and-excitation networks. Proceedings of the IEEE conference on computer vision and pattern recognition; 2018.

23. Woo S, Park J, Lee J-Y, Kweon IS, editors. Cbam: Convolutional block attention module. Proceedings of the European conference on computer vision (ECCV); 2018.

24. Wang L, Wang Y, Liang Z, Lin Z, Yang J, An W, et al., editors. Learning parallax attention for stereo image super-resolution. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2019.

25. Wang L, Guo Y, Wang Y, Liang Z, Lin Z, Yang J, et al. Parallax attention for unsupervised stereo correspondence learning. 2020. https://doi.org/10.1109/TPAMI.2020.3026899 PMID: 32976095

26. Nair V, Hinton GE, editors. Rectified linear units improve restricted boltzmann machines. Icml; 2010.

27. Wang Y, Ying X, Wang L, Yang J, An W, Guo Y, editors. Symmetric parallax attention for stereo image super-resolution. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2021.

28. Ying X, Wang Y, Wang L, Sheng W, An W, Guo YJISPL. A stereo attention module for stereo image super-resolution. 2020; 27:496–500.

29. Chen C, Qing C, Xu X, Dickinson PJIToM. Cross Parallax Attention Network for Stereo Image Super-Resolution. 2021.

30. Liu Y, Cheng M-M, Hu X, Wang K, Bai X, editors. Richer convolutional features for edge detection. Proceedings of the IEEE conference on computer vision and pattern recognition; 2017.

31. Xie S, Tu Z, editors. Holistically-nested edge detection. Proceedings of the IEEE international conference on computer vision; 2015.

32. Song X, Zhao X, Fang L, Hu H, Yu YJJCoCV. Edgestereo: An effective multi-task learning network for stereo matching and edge detection. 2020; 128(4):910–30.

33. Song X, Zhao X, Hu H, Fang L, editors. Edgestereo: A context integrated residual pyramid network for stereo matching. Asian Conference on Computer Vision; 2018: Springer.