A Learnable Region Support for Stereo Matching

Anonymous CVPR submission

Paper ID 000

Abstract

Despite deep neural networks significantly improve the performance of stereo matching, the challenge to handle outliers on occluded and indistinguishing areas still remains. Most current works leverage neural networks to improve the cost computation, while less attention is paid to cost aggregation and refinement. In this paper, we propose a learnable region support to guide the cost aggregation and refinement for stereo matching. In particular, the region support is modeled from the perspective of learning and favorably achieved by a novel depth segment network. With the learnable region support, we redesign the cost aggregation and refinement for stereo matching. Compared with the hand-crafted region support, our approach can more effectively handle the outliers. The experiments demonstrate that our learnable region support is superior to the state-of-the-art methods.

1. Introduction

Stereo matching has aroused considerable research interests in computer community. It aims to gain the accurate depth information by fusing two-frame images recorded by a stereo camera and exploiting the disparity between them. The disparity map can be widely used for 3D scene reconstruction, robotics, autonomous driving and virtual reality [3, 13, 14, 10]. Benefitting from the effectiveness of deep neural networks, many stereo matching have reached appealing performance with the pipeline composed of cost computation, cost aggregation, and refinement [30, 31, 8]. Most existing methods focus on cost computation [6, 20], while less attention is paid to cost aggregation and refinement. In stereo matching, actually, cost aggregation and refinement are indispensable to remove outliers[1], especially for the occluded and indistinguishable areas [30, 18, 22].

In general, cost aggregation and refinement are employed to rectify outliers by aggregating values among a specific region in stereo matching. Many studies [26, 25, 17] demonstrated that the region support can offer suitable regions and adaptive weights such as segment-based methods [11, 21] or cross-based methods [33, 17]. However, these hand-crafted methods are inferior to hold a basic assumption which pixels within the same region are supposed to share the same disparity value. In this paper, we propose a learnable region support favorably achieved by a novel depth segment network, which fully satisfies the assumption mentioned above. Furthermore, we reformulate both the cost aggregation and refinement with the guidance of our region support for stereo matching.

To ensure the learnable region support is composed of pixels at same disparity, the network should assign each pixel with a certain label, which is a typical pixel-wise labeling task [23]. In particular, we design a depth segment network by leveraging the fully convolutional network, in which we treat the disparity of each region as the training label. The support regions can be obtained from the segmentation results of the network. With support regions, the adaptive weights can further be computed according to deep representations and spatial relationships.

With the learnable region support, we consider the cost aggregation as the process of finding and rectifying outliers. In this work, we directly determine outliers by a simple variance measurement on each support region and rectify them by adaptive weights. The cost aggregation based on the hand-crafted region support can only be performed at the same disparity [15, 22]. In contrast, our cost aggregation can be carried out among different disparities by the learnable region support. Our learnable region support implies the mapping relationship of the disparity map between the low-resolution space and the high-resolution space. Therefore, it can also be employed for the up-sampling refinement of stereo matching. The up-sampling refinement can be naturally carried out by computing the disparity value at high-resolution according to the mapping relationship.

The stereo matching based on our learnable region support is performed through three key components including the generation of the learnable region support, cost aggregation and up-sampling refinement based on the region support. The whole framework of our stereo matching is shown

[1]Unless otherwise specified, we treat the mismatching value on cost volume and the error disparity on disparity map as outliers.

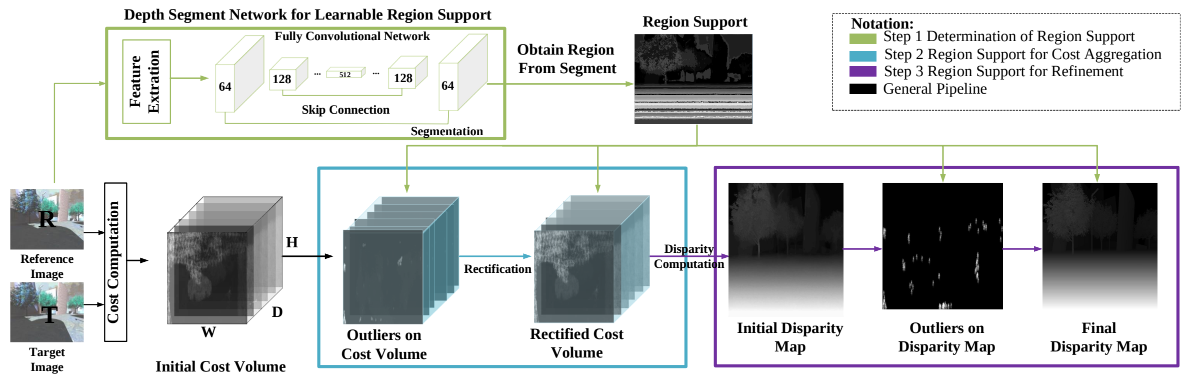


Figure 1. The Region Support Method for Stereo Matching. The region support for stereo mathcing can mainly be carried out in three steps: determining learnable region support, conducting cost aggregation according to the region support and refining the disparity with the region support.

in Figure.1. This pipeline can be compatible with any cost computation methods using neural networks. We evaluate our model on representative datasets (e.g., Scene Flow [10] and KITTI [3, 14]), which clearly demonstrates that the proposed approach is superior to the state-of-the-art methods. In summary, our contributions are three-fold.

- We propose a learnable region support which can effectively raise the accuracy of stereo matching. This work is, to the best of our knowledge, the first to model the region support from the perspective of learning.
- To effectively obtain the learnable region support, we design a novel depth segment network to obtain support regions composed of pixels at the same disparity, which fully satisfies the fundamental assumption of the region support.
- We redesign the cost aggregation and refinement for stereo matching with the guidance of learnable region support.

2. Related Work

2.1. Learnable Region Support

The region support concept is widely used in stereo matching [26, 25, 17, 15]. There are mainly two approaches to improve region support [15, 22], i.e., the generation of suitable regions and the computation of adaptive weights. Suitable regions can be achieved by sliding predefined windows [5, 32, 22] or designing adaptive shapes [15, 27]. The adaptive weights can be obtained according to the color intensity and spatial relationship of pixles among the region [12, 9]. Among these approaches, the segment-based region support attracts much attention because it provides both the effective support regions and adaptive weights [29, 12, 21]. Some of them simply use color intensity as the segment label [28, 33], or adopt a more complex score scheme to improve the segment [23, 15]. The adaptive weights can be obtained according to segmentation results. These methods are limited by the hand-crafted mechanism to main,tain the segmented regions composed of pixels at same disparity. The adaptive weights computed from the color intensity might not reflect the true similarity in the disparity space. Compared to [28, 21] which assume the support region is coincident to a homogeneous color, we enhance the region to be consistent with a certain disparity with the help of a learning mechanism. The learnable region support ensures that support regions consist of pixels at same disparity. The adaptive weights are also improved by the similarity computed from deep representation.

2.2. Stereo Matching using Neural Networks

Driven by the emergence of neural networks, stereo matching based on deep learning has proven to perform remarkably well on benchmark datasets. The usage of neural network is firstly introduced by Lecun et al. [30, 31] for cost computation. Then it is efficiently improved by Luo et al. [8]. Some current works further improve the cost computation by designing a more effective network to generate the robust representation [20, 16]. Meanwhile several works focused on the similarity measurement to compute matching cost [6, 19]. Many efforts have been devoted to the cost computation using neural networks, while less attention is paid to cost aggregation and refinement. To remove outliers on occluded and textureless areas, the learning mechanism for cost aggregation and refinement is essential. In this paper, we propose a depth segment network to enhance the region support concept for cost aggregation and refinement of stereo matching. Compared with the usual segmentation networks which utilize certain semantics as the label of each category [12, 7], we employ the disparity as the label for each region. The segmentation results can produce the learnable region support to carry out the cost aggregation and refinement.

3. Learnable Region Support

3.1. Problem Formulation

In this paper, the learnable region support is applied as the guidance to carry out the cost aggregation and refinement for stereo matching. A general stereo matching problem can be formulated as

$$M(x, y) = \max(s(R(x, y), T(x + d, y))), \qquad (1)$$

where M is the disparity map, $\max(\cdot)$ indicates the winner-take-all (WTA) strategy for disparity computation, $s(\cdot)$ is the similarity measurement for cost computation, R is the reference image, T is the target image and d is the disparity. In the past decade, we have witnessed substantial progress in stereo matching based on Eq. (1). However, outliers on cost volume and refinement are unfavorable for achieving the highly accurate disparity map. The outlier value on cost volume can be suppressed by improving the cost computation [8, 20] or rectifying mismatching values [33].

With the development of deep learning, many efforts are paid on the cost computation, however, outliers in the textureless and occluded areas are still unavoidable. As a result, the cost aggregation and refinement are essential to rectify outliers. The underlying hypothesis to employ the cost aggregation and refinement is that the value on either cost volume or disparity map should be similar for pixels at the same disparity, so outliers can be rectified according to the correct values. Under this hypothesis, the region support concept introduced in [26, 25], which focuses on the determination of the regions consisting of pixels at same disparity. With merits of support regions, adaptive weights for aggregation can be easily computed according to the color information and spatial relationship between pixels. Then the rectification on cost volume or disparity map can be achieved by aggregating the information among the support region using adaptive weights. However, the empirical hand-crafted mechanism [28, 1] is difficult to characterize the reliable information of the disparity space due to the naive usage of low-level color and spatial information.

To this end, we propose a novel learnable region support mechanism, in which the satisfactory disparity accuracy is obtained by determining the reliable regions comprised of pixels at the same disparity and enhancing adaptive weights with deep representations simultaneously. The learning mechanism equips the region support with the ability to model real information in the disparity space. Motivated by the segment-based region support [11, 23] which assumes that the reference image can be divided into a set of non-overlapping regions, where the region label is coincident with a specific color segment, we propose a depth segment network to enforce the segmentation. The region label is imposed on the disparity, which means that pixels in the same region would share same disparity. Therefore,

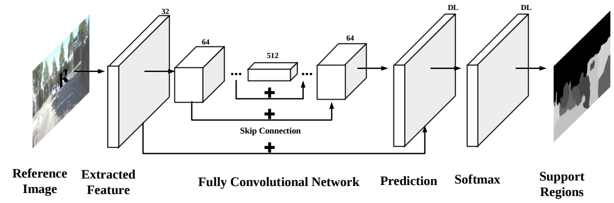


Figure 2. The depth segment network, each layer represents a 3 layres residual unit with relu and batch normalized function.The sub-sampling operation is at the strat of each residual unit and the layer before up-sampling is skip connected with the last layer before sub-sampling with same shape.

the learnable region is extremely useful for cost aggregation and refinement.

The definition of learnable segment-based region support can be expressed as

$$\begin{cases} \bigcup\limits_{i=1}^{n} \mathcal{N}_i = I \wedge \forall \mathcal{N}_i \bigcap \mathcal{N}_j = \varnothing \\ \qquad I(X) \in \mathcal{N}_{d=M(X)} \\ w(X, Y) = W\left(M(X) - M(Y)\right) \end{cases} \qquad (2)$$

Here, $\mathcal{N}$ denotes the set of support region, according to assumption of segment-based, the support regions are non-overlapped. Each pixel $I(X)$ on the postion $X = (x, y)$ of image I should belong to a certain region according its disparity. As for the adaptive weight w, the computation function W should be computed based on the similarity of the disparities between $M(X)$ and $M(Y)$. The learning of region support can be seen as the process to label each pixel with a particular disparity which is a typical segmentation problem. The learned support regions obtained from depth segment network are composed of pixels at the same disparity, which can find out outliers on cost volume and disparity map. Also, the adaptive weights can also be obtained by the deep representation and spatial relationship of pixels at the same region.

3.2. Depth Segment Network

As discussed above, our region support is composed of pixels at the same disparity. It is not necessary to ascertain the exact disparity for each pixel, therefore, we only need to find out which region the pixel belongs to, in other words, each pixel should have a label indicating its corresponding region. In this section, we propose a depth segment network which can divide the reference image into L regions, where L is a hyper-parameter for the number of support regions.

Many works [2, 23, 24] show that the single image depth prediction network is capable of obtaining the depth information from the reference image. Although the prediction may not be used to generate the accurate disparity map, it is sufficient to offer a trustworthy guidance for the region support determination. To obtain the region support, each pixel should be given a certain label representing the region disparity, which is a typical pixel-wise labeling task. The fully convolutional network on similar pixel-wise task such as semantic segment [24, 7] proves its effectiveness for this task. Therefore, we leverage this architecture to design the depth segment network and treat the disparity as the region label. The network is trained to annotate label each pixel with its region label which represent the disparity of pixels. The ground truth is obtained from the stereo matching disparity map with a simple threshold segment on the disparity map.

The proposed network can be divided into two parts: feature extraction and region prediction. The feature extraction part is composed of 8 residual units and S optional sub-sampling layers, where S is the scale ratio for the cost computation. Based on the extracted feature $E \in \mathbb{R}^{W/S \times H/S \times F}$ of the input image, we leverage the fully convolutional network to effectively perform pixel-wise labeling to obtain the depth information. The architecture of our network is illustrated in Figure.2 and the parameter setting can be found in Supplementary Material due to space limit.

The prediction step is to determine which region the pixel belongs to. Instead of using a simple classification strategy for segmentation, we formulate the problem as a regression process. Because the classification strategy for semantic segmentation assumes that the labels are irrelevant, but our label is actually relevant. Inspired by the GC-Net [6], we introduce the soft-argmin function expressed as

$$P(X) = 1/L \times \sum_{d=0}^{L} d \times \sigma(-E_l). \quad (3)$$

Here $P(X)$ is the predicted region label of each pixel, L represents the hyper-parameter for the number of support regions, E_l stands for the output of the last layer of the network with size of $W \times H \times L$ and $\sigma(\cdot)$ denotes the softmax operation along L dimension. Then the loss can be given by the following supervised regression loss:

$$Loss = 1/(W \times H) \times \sum_{i=0}^{W \times H} |P(X) - G(X)|. \quad (4)$$

Here, $G(X)$ is the region label at ground truth and the $|\cdot|$ represents the absolute value. The comparison of regression loss and classification loss is conducted in Figure.4, where we can see the regression loss leads to a more stable training process.

Compared to the general classification operation with a certain label, the regression loss can provide the similarity of the pixel to a certain region. According to the predicted

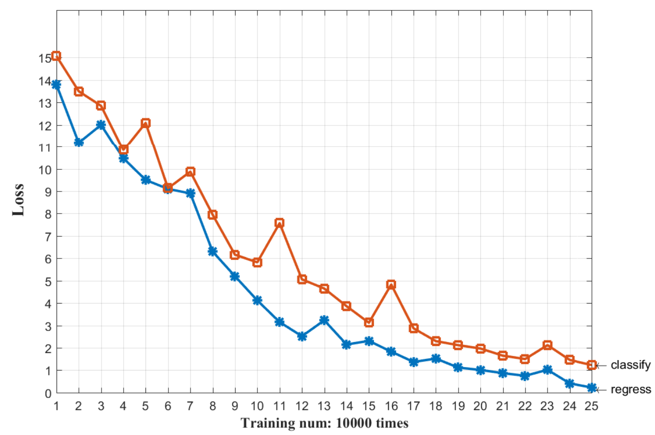


Figure 3. The comparision of regression loss and classification loss on Scene Flow dataset.

value, we can obtain the initial support region by labeling each pixel with its region number:

$$P(X) = \lfloor P(X) \rfloor, \quad (5)$$

where the $\lfloor \cdot \rfloor$ represents the round function.

The obtained initial region support from the segmentation result might be too large to reach a high accuracy, therefore, we segment the initial region support to smaller regions. The final region support is constrained to contain no more than 100 pixels in each support region. The final segmentation operation is carried out according to the segmentation results and spatial relationship. The details are shown in Algorithm.1.

Algorithm 1: Obtain Region Support

```
Input: Segmentation Results from netowrk P ∈ R^{W×H}
Output: The Region Support N
Step1: Obtain the Initial Region Support from Segmentation
N = ∅
P = round(P)
while ∃X ∉ N do
    add X to N_{P(X)}
end
Step2: Segment the Initial Region Support to Small Region
N_s = ∅
for i=0:len(N) do
    while pixel X in N(i) and X not in N_s do
        N_t = ∅
        add X to N_t
        while pixel Y in N(i) and Y not in N_s do
            if |X − Y|^2 < 11 then
                add Y to N_t
                if len(N_t)> 100 then
                    add N_t to N_s
                    Break
                end
            end
        end
    end
end
Return N = N_s
```

Algorithm 2: Region Support Cost Aggregation

Input: The initial cost volume $V \in \mathbb{R}^{W \times H \times D}$, Region Support $\mathcal{N}$, Deep Feature $F \in \mathbb{R}^{W \times H \times 32}$
Output: The aggregated cost volume $A \in \mathbb{R}^{W \times H \times D}$

Step 1: Finding outliers
$O = \varnothing$
for i *=0:len(*$\mathcal{N}$*)* **do**
    mv=1/len($\mathcal{N}(i)$) $\times \sum \sigma^2(V(\mathcal{N}(i)))$
    if X *in* $\mathcal{N}(i)$ *and* $\sigma^2(V(X)) > mv$ **then**
        add X to O
    end
end
Step 2: Rectifying outliers
for o *in* O **do**
    $\mathcal{N}_o = o \subset \mathcal{N}$
    $A(o) = (\sum \cos(F(o), F(\mathcal{N}_o)) \times dis(o, \mathcal{N}_o) \times V(o))/len(\mathcal{N}_0)$
end
Return A

4. Learnable Region Support for Stereo Matching

In this section, we offer applications of our learnable region support on cost aggregation and sub-sampling refinement for stereo matching.

4.1. Learnable Region Support for Cost Aggregation

With the learnable region support, we reformulate the cost aggregation as the problem of determining and rectifying outliers on the cost volume. As followed by [lecun], cost computation can extract the deep representation for each pixel of stereo pair images by a Siamese network, which can be formulated by a function $f : \mathbb{R}^{W \times H \times C} \to \mathbb{R}^{W \times H \times F}$, where W, H, C are the width, height and channel of input images and F represents the channel of the feature map. To compute the matching cost of each pixel in the reference image, the similarity measurement is employed , which can be formulated as $s : \mathbb{R}^{W \times H \times F} \times \mathbb{R}^{W \times H \times F} \to \mathbb{R}^{W \times H \times D}$, where D represents the number of disparity levels. The whole cost computation can be formulated as the function $c : \mathbb{R}^{W \times H \times C} \times \mathbb{R}^{W \times H \times C} \to \mathbb{R}^{W \times H \times D}$. Given a stereo image pair: $R, T \in \mathbb{R}^{W \times H \times 3}$, where R and T represent the reference and target image, respectively. The cost volume V is calculated by

$$V = c(R, T) = s(f(R), f(T)). \tag{6}$$

The obtained initial cost volume by Eq.(6) still has some incorrect matching value on textureless and occluded areas, so the cost aggregation method step is critical to rectify. The general region support methods perform the aggregation operation to all pixels among the support region by integrating other pixels in the same region with adaptive weights. A typical region support cost aggregation method is formulated as

$$A(X, d) = \sum_{Y \in \mathcal{N}(X)} w(R(X), R(Y)) \times V(Y, d). \tag{7}$$

Here, $\mathcal{N}(X)$ is the support region for the pixel $R(X)$, $w(\cdot)$ is the adaptive weight, $V(Y, d)$ is the value of initial cost volume and $A(X, d)$ is the aggregated cost value. From the depth segment network, we can obtain $\mathcal{N}(X)$ for each pixel $R(X)$ on the reference image. The outliers are found by a variance measurement:

$$O = \sigma^2(V(X)) > \sigma^2(\mathcal{N}(i)). \tag{8}$$

Here $\sigma^2(\mathcal{N}(i))$ is the mean variance of region i on the d disparity level of the cost volume, and $\sigma^2(X)$ is the variance for the matching value $V(X, d)$.The computation of aggregation weights will be computed between outliers and other pixels one by one:

$$w(X, Y) = s(f(R(X)), f(R(Y))) \times dis(X, Y). \tag{9}$$

Here,$s(\cdot)$ represents the similarity measurement and $dis(\cdot)$ represents the spatial relationship of pixles. We leverage the deep feature from cost computation as the representation for each pixel in the process of similarity computation. For the similarity measurement, we use an inner-product layer like Content-CNN [8] to compute the inner product as the weights for aggregation. The $dis(\cdot)$ spatial relationship between pixels is defined as

$$dis(X, Y) = \begin{cases} \lambda_1 & 0 \leq if|X - Y|^2 < 1 \\ \lambda_2 & 1 \leq if|X - Y|^2 < 2 \\ \lambda_3 & 2 \leq if|X - Y|^2 < 3 \\ \lambda_4 & 3 \leq if|X - Y|^2 \end{cases} \tag{10}$$

where $|\cdot|^2$ represents the Euclidean distance between two pixels. In this paper, we set λ_1=0.75, λ_2=0.5, λ_3=0.25 and λ_4=0.1. After getting the regions and weights, the cost aggregation can be conducted by Eq.(7). The detailed algorithm can be carried out as Algorithm.2.

The learnable region support can lead to aggregation among different disparity level. After completing the aggregation among the same disparity, the aggregation along different disparity is carried out as

$$A(X, d) = \sum_{d \in \mathcal{N}(X)} w(T(X), T(Z)) \times V(Z, d), \tag{11}$$

where, $\mathcal{N}(X)$ represents the support region for the pixel $T(Z)$ on the target image. The cost aggregation on the same disparity assumes that the cost values for pixles in the same support region on reference image should be the same on the cost volume. In contrast, the assumption for cost aggregation among different disparity is that the cost values of pixles in the same support region on the target image should be same. The reason is that the $V(x, y, d)$ in the cost volume means the matching cost between $R(x, y)$ and $T(x + d, y)$, so if $T(x + d, y)$ and $T(x + d, y)$ are similar, then $V(x, y, d)$ should be similar with $V(x, y, d + i)$. With this formulation, the aggregation can be carried out according to the region support from the target image. Each row of the support region determines $\mathcal{N}(X)$ for pixles in this row. For $X = (x, y)$ and $Y = (x + d, y)$, the similarity measurement is the same as weights computation in the same disparity, the weights can be formulated as

$$w(T(X), T(Y)) = s(f(T(X)), f(T(Y))) \times dis(d). \quad (12)$$

Algorithm 3: Region Support Refinement

Input: The low-resolution disparity map $LD \in \mathbb{R}^{W/S^2 \times H/S^2}$ the region support $\mathcal{N}$
Output: The origin-resolution disparity map $HD \in \mathbb{R}^{W \times H}$

$HD = zeros(W, H)$
$HD(round(X \times S^2/2)) = S^2 \times LD(X)$
while $\exists HD(X) == 0$ **do**
  while Y *in the same* $\mathcal{N}$ *with* X **do**
    $\mathcal{N}_h = \varnothing$
    if *HD(Y)!=0* **then**
      H
    end
    add Y to $\mathcal{N}_h$
  end
  $HD(X) = (\sum \cos(F(X), F(\mathcal{N}_h)) \times HD(\mathcal{N}_h))/len(\mathcal{N}_h)$
end
Return HD

4.2. Learnable Region Support for Refinement

Benefiting from learnable region support, the refinement to rectify outliers can be reformulated as

$$M(X) = \sum_{Y \in \mathcal{N}(X)} w(X) \times M^{'}(Y), \quad (13)$$

where $M^{'}$ represents the initial disparity image and M is the refined image. This operation can be easily conducted like the cost aggregation among the same disparity in Algorithm.2.

Furthermore, the learnable region support allows the cost computation and cost aggregation to work in the low-resolution space, which can remarkably reduce the computation burden. The increase of down-sampling scaling parameter S can reduce S^2 times on the computation resource. However, the down-sampling results in a low-resolution disparity map, which is insufficient for the high accuracy of stereo matching. In this paper, we propose a solution to carry out the up-sampling for the low-resolution disparity map with our learnable region support.

We can obtain the original resolution region support by removing the optimal sub-sampling layers from the depth segment network. The low-resolution disparity map $J \in \mathbb{R}^{W/S \times H/S}$ can be obtained from the disparity computation. Based on J of low-resolution and the learnable region support $\mathcal{N}$ of original-resolution, the up-sampling refinement can be reformulated as

$$K(Y) = \sum_{Y \in \mathcal{N}(X)} S^2 \times m(X, Y) \otimes J(X), \quad (14)$$

where K represents the original resolution image, $m(\cdot)$ is the mapping weights during up-sampling, $\mathcal{N}(X)$ denotes the area mapping into the high-resolution space and $\otimes$ stands for the up-sampling operation. We assume that each pixel in the low-resolution space lies in the center of the up-sampled region in the high-resolution space, therefore $Y = S^2 \times X/2$. Each support region can be determined by up-sampled pixels from the low-resolution space which means pixles among a specific region in the high-resolution space can be determined by a few pixels on the low-resolution space. Because we can confirm the relationship between each pixel in the support region, so the mapping relationship for up-sampling can be obtained simply by the adaptive weight computation in Eq. (9). Since the resolution is up-sampled S^2 times, the mapping value should be multiplied by S^2. The whole process of sub-sampling refinement is shown in Algorithm.3.

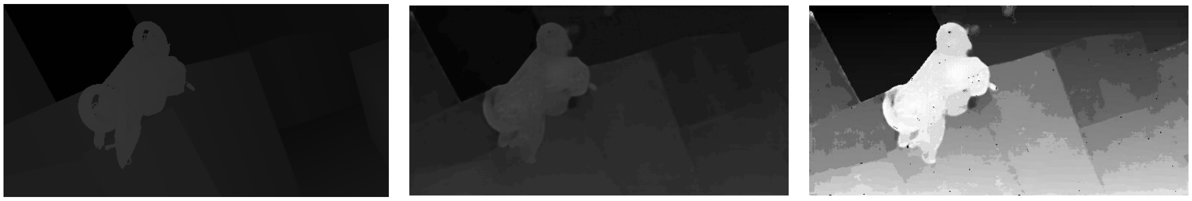
5. Experiment Evaluation

In this section, we present the qualitative and quantitative results of our learnable support on Scene Flow [10] and KITTI [3, 14] benchmarks. We reproduce the state-of-the-art GC-Net as the baseline to evaluate our learnable region support. The experiments are implemented with Tensor Flow and trained with standard RMSProp method. The depth segment network is trained for 180K iterations on Scene Flow and fine-tuned on KITTI for 40K times using a constant learning rate of 0.001, respectively. The training on Scene Flow takes 22 hours on an NVIDIA 1080TI.

5.1. The Evaluation for Stereo Matching

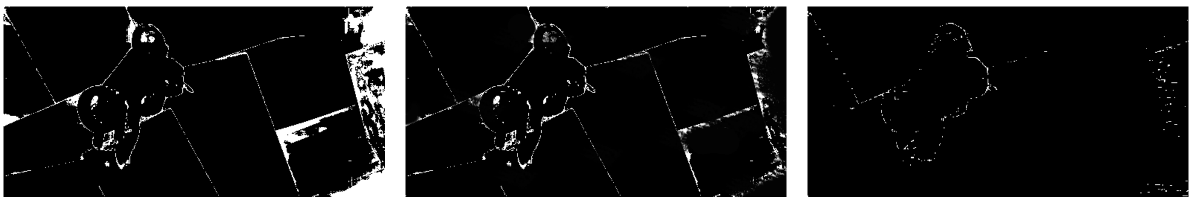
We employ the learnable region support to rectify the cost volume and disparity map of the GC-Net, respectively. The depth segment network is trained on Scene Flow and KITTI. The Scene Flow dataset contains 35454 training and 4370 testing images, which is large enough to train the depth segment network without overfitting. In addition, the ground truth of this synthetic dataset is dense and has few erroneous labels. The KITTI dataset contains 194 training and 195 test image pair consist of images of challenging and varied road scene obtained from LIDAR data. The shortcoming is that the quantity of KITTI dataset is not large enough to train the neural network.

We compare our region support based stereo matching method of state-of-the-art methods by the model pre-trained



Ground Truth **a** **b**

Figure 4. The visualization of learnable region support and the result of depth segment network. There are L regions in image a and more than 2000 regions on image b. Each of the region in b contains no more than 100 pixels. Compared to the ground truth, we can see the image a and image b capture the real depth information.



Outliers of Ground Truth **Outliers Detected by Region Support** **Outliers Rectified by Region Support**

Figure 5. The effectivness of region support for refinement. We can see the region support can effectively detect and rectify the outliers.

Table 1. Comparisons on KITTI2015

Model	All pixels			Non-Occluded Pixels			Time(s)
	D1-bg	D1-fg	D1-all	D1-bg	D1-fg	D1-all	
GC-Net[6]	2.21	6.16	2.87	2.02	5.58	2.61	0.9
MC-CNN[30]	2.89	8.88	3.89	2.48	7.64	3.33	67
Displetv v2[4]	3.00	5.56	3.43	2.73	4.95	3.09	265
PSMNet	**1.97**	4.41	**2.38**	**1.81**	4.00	**2.17**	1.3
L-ResMatch[20]	2.72	6.95	3.42	2.35	5.74	2.91	48
iResNet	2.35	**3.62**	2.56	2.18	**3.09**	2.33	0.1
Our model	**2.07**	**4.25**	**2.49**	**1.97**	**5.24**	**2.27**	**30.2**

Table 2. The Comparisons of Scene Flow with GC-Net

Model	error>1px	error>3 px	error>5 px	Time(s)
GC-Net	16.9	9.34	7.22	**0.95**
Baseline	17.22	9.73	7.31	1.13
Cost Aggregation	**13.98**	**7.46**	6.24	28.53
Refinement	14.46	7.51	**6.15**	1.36
Ours	**12.87**	**7.26**	**5.72**	29.42

Table 3. The ability to handle the outliers.

Method	Scene Flow			KITTI		
Benchmark	Outliers	Found	Rectified	Outliers	Found	Rectified
Cost Aggregation	16860	4936	3847	3254	4307	1544
Refinement	16860	4379	3724	3254	3249	1324

on Scene FLow and fine-tuned on KITTI. The results are shown at Table.1. We can see the learnable region support raise the accuracy of baseline GC-Net by 9.3% and also reach the state-of-the-art performance on KITTI. The promotion of all pixels is much remarkable than the promotion on non-occluded pixels, which proves that our learnable region support can effectively rectify the outliers on occluded areas.

Furthermore, We test the effectiveness of our cost aggregation and refinement on Scene Flow. The results in Table.2 shows that learnable region support for cost aggregation is good at rectifying outliers with small error because the promotion on 1 px error is more obvious than 3 px and 5 px. The reason lies that the rectification of outliers on cost aggregation can lead to a continuous disparity map through the proposed soft-argmin function. In contrast, the refinement performs better on outliers with big error because the rectification on disparity map offers truncate disparity value. The combination of cost aggregation and refinement can lead to a more balanced result among all pixel errors. We can see the computation time is mainly used for the cost aggregation since the cost aggregation at different disparities is carried out separately.

5.2. The Analysis of Learnable Region Support

In this section, we present a further evidence to analyze the effectiveness the learnable region support can be effective for the cost aggregation and refinement.

Depth Segment Network We test the ability of our depth segment network to reliably provide the depth information. In Figure.4(a), we visualize the segment result of depth segment network. We can see that the image is coarsely segmented into L regions according to the true disparity, which means our depth segment network obtains the reliable depth information from the input image. Then to obtain a more fitness result, the segment results are divided into small regions. The obtained support regions are shown in Figure.4(b). Compared to Figure.4(a), there are thousands region in the final support regions, each of the region contains no more than 100 pixels. From Figure.4, we can see the learnable region support realizes the basic assumption for region support, in which each region is composed of pixels at same disparity.

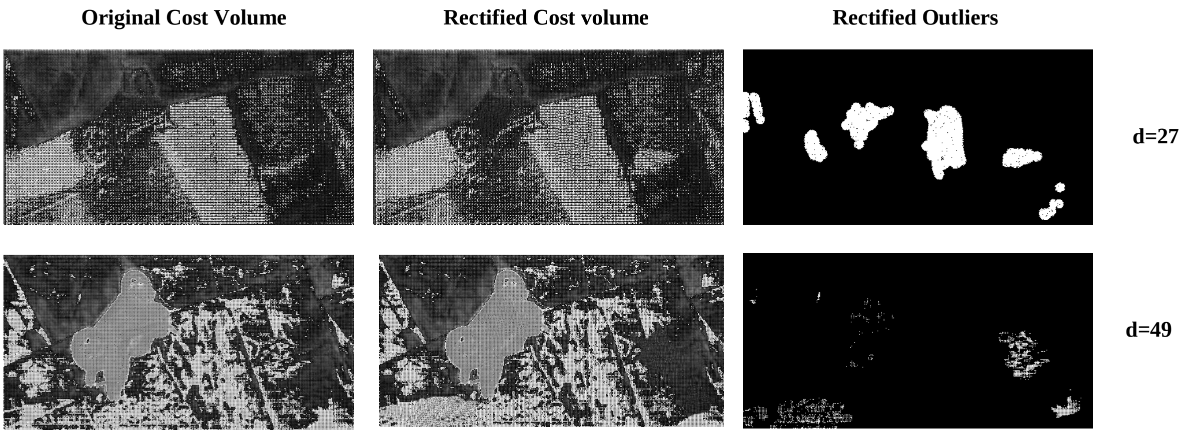


Figure 6. The visualization of cost volume at disparity 27 and 49. We can see the region support helps to aggregate the value at cost volume.

Figure 7. The quality of up-sampling refinement.

For Cost Aggregation We test the region support for the cost aggregation by qualitatively visualizing the aggregated results on cost volume and quantitatively analysis the final result on disparity. In Figure.6, we compare the original cost volume with the rectified result on disparity 27 and 49. From the visualization of outliers, we can see the region support offers different guide for the cost aggregation in different disparity. In Table.3, we quantify the outliers and rectification results. We can see the learnable region support can correctly find more than 32.72% percent of the outliers and rectify 26% of them. The rectified cost volume can approximately lead to an average 18% promotion for the final disparity map. These remarkable experiment results prove that the region support is effective for the cost aggregation.

For Refinement We use the learnable region support for the normal refinement work. The obtained outliers and rectified results are shown in Figure.5, while the quantification is shown in Table.3. We can see the learnable region support can find more than 32% outliers on the disparity map and rectify more than 75% of outliers on Scene FLow. The learnable region support leads to a 18% promotion for the disparity. To test the effectiveness of the up-sampling refinement, the result is shown in Figure.7. The up-sampled disparity is able to keep the fitness and maintain high accurate of the disparity map. The results prove that the mapping relationship obtained the region support can effectively handle the up-sampling work.

6. Conclusion

In this paper, we have presented a learnable region support for stereo matching. The depth segment network was designed to carry out the learning mechanism for region support. With the learnable region support, the cost aggregation and refinement were redesigned for stereo matching. We have demonstrated that learned region support can fully satisfy the basic assumption for region support from the experiments. With the analysis on cost aggregation and refinement, we have proved that the learnable region support is effective for stereo matching. For example, the cost aggregation with region support rectified more than 18% outliers on the disparity map based on GC-Net. The stereo matching method with the redesigned cost aggregation and refinement finally achieved the state-of-the-art performance both on Scene Flow and KITTI.

References

[1] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. *International Journal of Computer Vision*, 92(1):1–31, 2011. 3

[2] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in neural information processing systems*, pages 2366–2374, 2014. 3

[3] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3354–3361. IEEE, 2012. 1, 2, 6

[4] F. Guney and A. Geiger. Displets: Resolving stereo ambiguities using object knowledge. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4165–4175, 2015. 7

[5] J. Ha, J. Jeon, G. Bae, S. Jo, and H. Jeong. Cost aggregation table: cost aggregation method using summed area table scheme for dense stereo correspondence. In *International Symposium on Visual Computing*, pages 815–826. Springer, 2014. 2

[6] A. Kendall, H. Martirosyan, S. Dasgupta, P. Henry, R. Kennedy, A. Bachrach, and A. Bry. End-to-end learning of geometry and context for deep stereo regression. 2017. 1, 2, 4, 7

[7] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015. 2, 4

[8] W. Luo, A. G. Schwing, and R. Urtasun. Efficient deep learning for stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5695–5703, 2016. 1, 2, 3, 5

[9] Z. Ma, K. He, Y. Wei, J. Sun, and E. Wu. Constant time weighted median filtering for stereo matching and beyond. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 49–56, 2013. 2

[10] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. 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. 1, 2, 6

[11] G. Medioni and R. Nevatia. Segment-based stereo matching. *Computer Vision, Graphics, and Image Processing*, 31(1):2–18, 1985. 1, 3

[12] X. Mei, X. Sun, W. Dong, H. Wang, and X. Zhang. Segment-tree based cost aggregation for stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 313–320, 2013. 2

[13] X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang, and X. Zhang. On building an accurate stereo matching system on graphics hardware. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 467–474. IEEE, 2011. 1

[14] M. Menze and A. Geiger. Object scene flow for autonomous vehicles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3061–3070, 2015. 1, 2, 6

[15] D. Min, J. Lu, and M. N. Do. A revisit to cost aggregation in stereo matching: How far can we reduce its computational redundancy? In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1567–1574. IEEE, 2011. 1, 2

[16] J. Pang, W. Sun, J. Ren, C. Yang, and Q. Yan. Cascade residual learning: A two-stage convolutional neural network for stereo matching. In *International Conf. on Computer Vision-Workshop on Geometry Meets Deep Learning (ICCVW 2017)*, volume 3, 2017. 2

[17] U. Raghavendra, K. Makkithaya, and A. Karunakar. Anchor-diagonal-based shape adaptive local support region for efficient stereo matching. *Signal, Image and Video Processing*, 9(4):893–901, 2015. 1, 2

[18] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International journal of computer vision*, 47(1-3):7–42, 2002. 1

[19] A. Seki and M. Pollefeys. Patch based confidence prediction for dense disparity map. In *BMVC*, 2016. 2

[20] A. Shaked and L. Wolf. Improved stereo matching with constant highway networks and reflective confidence learning. *arXiv preprint arXiv:1701.00165*, 2016. 1, 2, 3, 7

[21] F. Tombari, S. Mattoccia, and L. Di Stefano. Segmentation-based adaptive support for accurate stereo correspondence. *Advances in Image and Video Technology*, pages 427–438, 2007. 1, 2

[22] F. Tombari, S. Mattoccia, L. Di Stefano, and E. Addimanda. Classification and evaluation of cost aggregation methods for stereo correspondence. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. 1, 2

[23] D. Wang and K. B. Lim. Obtaining depth map from segment-based stereo matching using graph cuts. *Journal of Visual Communication and Image Representation*, 22(4):325–331, 2011. 1, 2, 3

[24] P. Wang, X. Shen, Z. Lin, S. Cohen, B. Price, and A. L. Yuille. Towards unified depth and semantic prediction from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2800–2809, 2015. 3, 4

[25] Z.-F. Wang and Z.-G. Zheng. A region based stereo matching algorithm using cooperative optimization. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. 1, 2, 3

[26] Y. Wei and L. Quan. Region-based progressive stereo matching. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1, pages I–I. IEEE, 2004. 1, 2, 3

[27] Q. Yang. A non-local cost aggregation method for stereo matching. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1402–1409. IEEE, 2012. 2

[28] Q. Yang, L. Wang, R. Yang, H. Stewénius, and D. Nistér. Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(3):492–504, 2009. 2, 3

[29] K.-J. Yoon and I. S. Kweon. Adaptive support-weight approach for correspondence search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):650–656, 2006. 2

[30] J. Zbontar and Y. LeCun. Computing the stereo matching cost with a convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1592–1599, 2015. 1, 2, 7

[31] J. Zbontar and Y. LeCun. Stereo matching by training a convolutional neural network to compare image patches. *Journal of Machine Learning Research*, 17(1-32):2, 2016. 1, 2

[32] K. Zhang, Y. Fang, D. Min, L. Sun, S. Yang, S. Yan, and Q. Tian. Cross-scale cost aggregation for stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1590–1597, 2014. 2

[33] K. Zhang, J. Lu, and G. Lafruit. Cross-based local stereo matching using orthogonal integral images. *IEEE transactions on circuits and systems for video technology*, 19(7):1073–1079, 2009. 1, 2, 3