IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 4, 2020, accepted October 5, 2020, date of publication October 9, 2020, date of current version November 2, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3029832

Compact StereoNet: Stereo Disparity Estimation via Knowledge Distillation and Compact Feature Extractor

QINQUAN GAO[1,2,3], YUANBO ZHOU[1,2], (Graduate Student Member, IEEE), GEN LI[3], AND TONG TONG[1,2,3]

[1]College of Physics and Information Engineering, Fuzhou University, Fuzhou 350108, China
[2]Fujian Key Laboratory of Medical Instrumentation and Pharmaceutical Technology, Fuzhou 350108, China
[3]Imperial Vision Technology, Fuzhou 350002, China

Corresponding author: Tong Tong (ttraveltong@gmail.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61901120 and Grant 61802065, and in part by the Science and Technology Program of Fujian Province of China under Grant 2019YZ016006.

ABSTRACT Stereo disparity estimation is a difficult and crucial task in computer vision. Although many experimental techniques have been proposed in recent years with the flourishing of deep learning, very few studies take into account the optimization of computational complexity and memory consumption. Most previous works take advantage of stacked 3D convolutional block to generate fine disparity, but with a high computational cost and a large memory consumption. Considering the aforementioned problem, in this paper, we proposed an efficient convolutional neural architecture for stereo disparity estimation. In particular, a compact and efficient multi-scale extractor named MCliqueNet with stacked CliqueBlock was proposed to extract the more refined features for constructing multi-scale cost volume. In order to reduce the computational cost and maintain the accuracy of disparity, we utilized knowledge distillation scheme to transfer contextual features from a teacher network to a student network. Furthermore, we present a novel adaptive $Smooth_{L1}$ (ASL) Loss for calculating the similarity between the contextual features of the teacher network and those of the student network, resulting in a more robust distillation process. Experimental results have shown that our method achieves competitive performance on the challenging Scene Flow and KITTI benchmarks while maintaining a very fast running speed.

INDEX TERMS Stereo disparity estimation, 3D convolution, knowledge distillation, compact extractor, cost volume.

I. INTRODUCTION

Estimating indoor and outdoor scenes via images is a challenging problem for 3D vision, which is due to the fact that the depth of information is lost in the process of capturing pictures. Therefore, it is crucial for 3D vision to accurately estimate the missing depth information from images. In general, depth estimation can be divided into active depth estimation and passive depth estimation. Due to the low cost of passive depth estimation, it is widely used in 3D vision. Furthermore, passive depth estimation can be divided into monocular depth estimation and stereo depth estimation, and stereo depth estimation usually works better. Stereo depth estimation uses the relationship between disparity and depth to estimate depth. The corresponding disparity is estimated by matching pixels from rectified image pairs captured by two cameras. The relationship for converting between depth and disparity is $D = Fl/d$, where D denotes depth, d denotes disparity, F denotes the focal length of the camera, l denotes the distance between two camera centers.

If the precisely disparity can be estimated, we can get the exactly depth. Therefore, stereo disparity estimation from a pair of stereo images has drawn more and more attention, which is also widely used in 3D reconstruction [2], [3], augmented reality (AR) [4], [5], self-driving [6], [7] and robotics [8]–[10]. The traditional stereo disparity estimation methods can be divided into global energy function [11], [12] and local similarity [13], [14]. These methods have several steps including matching cost computation, cost aggregation, disparity optimization and post-processing [15].

The associate editor coordinating the review of this manuscript and approving it for publication was Naveed Akhtar.

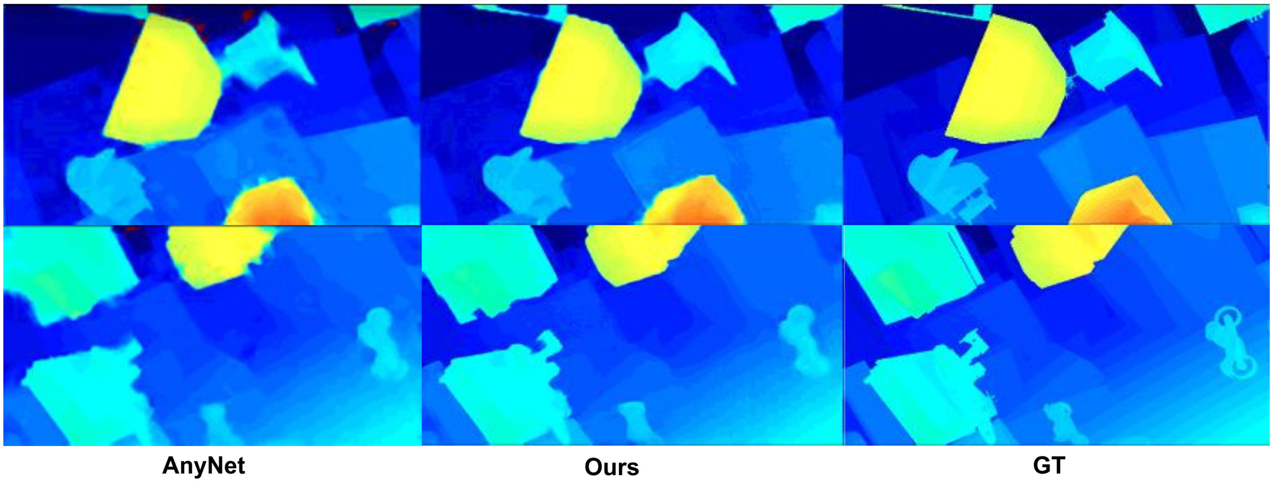


FIGURE 1. Comparing our proposed method with AnyNet [1] on scene flow dataset, it is clear that our results are better.

However, most traditional approaches are very sensitive to occlusions, textureless, reflective surface and thin structures areas.

In contrast, the emergence of powerful tool—Convolutional Neural Network (CNN) can extract useful information and contextual features from an image or video stream for stereo disparity estimation. Not only can it effectively improve feature matching accuracy, but also reduce the handicrafts. Zbontar *et al.* [16] first presented a CNN-based method to extract features and matching for estimating stereo disparity, which pioneered the application of CNN to feature matching and achieved remarkable results. Therefore, more researchers are using CNN to extract and match features to improve the accuracy of stereo disparity estimation. Although these methods achieve remarkable results with CNN features, they still fail in some challenging areas like occlusions.

To address aforementioned problem, methods based on 3D convolution [17], [18] were proposed to aggregate matching cost between two image features. They fully utilized the spatial information to achieve more accurate results. However, the calculation of 3D convolution will take heavy computation and memory burden. The process of 3D convolution would make training and deployment computationally expensive in practice.

In order to reduce the number of parameters in networks, the most intuitive way is to cut down on the number of 3D convolutions. However, reducing the number of 3D convolutions is usually at the expense of the accuracy. Therefore, the model compression of 3D convolutions without losing accuracy is a hot topic for researchers. For instance, GANet [19] and GwcNet [20] were proposed. However, these methods still cannot work in realtime due to the very high computational cost.

Hence, this study is dedicated to the development of realtime models. StereoNet [21] is one of them. It was proposed to provide a realtime implementation using a fully end-to-end CNN with a 720 × 1280 input on an Nividia Titan X GPU. Furthermore, AnyNet [1] was also proposed to effectively run on a computation-limited platform, which is a lightweight network, achieving state-of-the-art results while having much fewer parameters than StereoNet [21].

To further enable the model to be used in real-world scenes, we believe that model compression is of critical importance for disparity estimation. Model compression approaches can be used to speedup the inference process with less memory and calculation requirements. These approaches can be divided into network pruning [22], [23], parameter quantization [24]–[26], low-rank decomposition [27]–[29] and knowledge distillation (KD) [30]. In terms of both computational cost and memory resource, numerous studies based on knowledge distillation have been reported in computer vision and image processing. However, knowledge distillation was first proposed by [30] for classification task. It cannot be directly used to solve the regression problem in our task.

In this paper, to resolve aforementioned problem, we proposed a novel fully end-to-end architecture for stereo disparity regression using knowledge distillation scheme. Meanwhile, to reduce the affection of the normally used L2 loss by outliers in knowledge distillation module, we proposed a novel loss called adaptive $Smooth_{L1}$ (ASL) Loss. Furthermore, we combined Focal Loss (FL) [31] and ASL Loss to improve the performance of student network. In addition, a compact feature extraction network called multi-scale CliqueNet (MCliqueNet) was proposed to significantly improve the accuracy of disparity under a real-time condition.

Our main contributions can be summarized as follows:

1) We present a realtime framework in stereo disparity estimation that yields significant improvements over state-of-the-art results without increasing computational cost.
2) To our best knowledge, this is first work that utilizes distillation knowledge scheme for disparity regression.
3) We show that the direct use of knowledge distillation is hardly helpful in stereo disparity estimation, thus

we proposed an adaptive distillation method to guide the student network by using an adaptive loss function, which can alleviate the impact of bad teacher propagation.

4) A new network structure called MCliqueNet was proposed for feature extraction in stereo disparity estimation and its effectiveness has been demonstrated.

II. RELATED WORKS

Feature matching is an important step in both traditional and learning based algorithms in stereo disparity estimation. The first step in feature matching is to extract features. However, artificial features such as SIFT [32], SURF [33] and ORB [34] are often time-consuming and not robust in occasions, textureless, reflective surface and thin structures areas, resulting in many mismatches in the feature matching step. That is due to the fact that the wrong matches keep lower cost than correct matches. Therefore, CNN-based feature extraction neural networks are emerging. Meanwhile, CNN-based approaches not only improve the robustness of feature matching, but also leverage a variety of temporal and spatial information, making the performance of learning-based methods constantly stand on SOTA. Therefore, in this part, we will introduce traditional algorithms and the learning-based methods respectively in detail.

A. TRADITIONAL STEREO DISPARITY ESTIMATION

Stereo disparity estimation has been investigated over several decades, since the classic paper [35] was presented. The traditional stereo disparity estimation methods can be divided into global energy function [11], [12] and local similarity [13], [14]. For the methods based on local similarity, SAD (Sum of absolute differences), SSD (Sum of squared differences), NCC (normalized cross-correlation) [36] are used to calculate the local similarity. Although it can achieve dense disparity, it is very sensitive to outside interference. For the methods based on global energy function, the key is to construct energy functions and find a solution for optimization problems. Common ways to solve the optimization problems include Dynamic Programming [37], Graph Cut [38], and Neural network [39]. More comprehensive results have been reported in literature [15].

B. LEARNING-BASED STEREO DISPARITY ESTIMATION

Although there have been some breakthroughs with traditional methods on some complex conditions. For learning-based methods, which can be traced back to 2016, Zbontar and LeCun [16] first utilized a convolutional architecture to compute matching cost of the image patches. After that, Luo *et al.* [40] utilized a Siamese architecture to improve the accuracy. Mayer *et al.* [41] attributed a big synthetic Scene Flow dataset to promote an end-to-end training [17], [42]. Especially, GC-Net [17] was proposed, which is first work to use a 3D convolution to merge geometry and contextual information with *soft argmin* for disparity regression. Following GC-Net [17], PSMNet [18] was presented, which used a pyramid pooling module and stacked a 3D hourglass network in cost volume step to refine disparity map and obtained remarkable performance than previous related works. To improve the accuracy of disparity, Yang *et al.* [43] utilized a semantic feature for disparity prediction.

C. LIGHTWEIGHT AND REALTIME CNN FOR STEREO DISPARITY ESTIMATION

Despite the proliferation of CNN-based approaches, it poses a significant challenge for real-world applications. Due to the requirement of high computations of previous CNN-based approaches, they are not usable in some realtime applications. Thus, it is essential to develop a lightweight approach for stereo disparity estimation to fulfill the requirement of these realtime applications.

A lightweight network is a good choice for the trade-off between accuracy and computation. The related lightweight network includes pruning [22], [23], quantization(e.g. binary connect [24], XNOR-Net [25]), low-rank decomposition (e.g. mobilenet series [27], [44], [45], shufflenet series [28], [46]) and knowledge distillation(KD) [30]. These methods have been successfully embedded on a resource-limited platform with model compression techniques.

Lightweighting of stereo disparity estimation has been studied. Du *et al.* [47] utilized low-rank decomposition to extract features in stereo disparity estimation. Tulyakov *et al.* [48] used bottleneck modules to decrease the memory footprint in inference. Du *et al.* [47] adopted an efficient feature extractor with depth-wise separable convolutions to reduce computational cost. In addition, GANet [19] combined the traditional and deep learning methods to decrease the use of 3D convolutions by adding SGA and LGA layers for aggregating disparity. Guo *et al.* [20] also proposed a Group-wise Correlation Stereo Network, which utilized Group-wise cost volume to cut the computation cost. Duggal *et al.* [49] developed a differentiable PatchMatch module to speedup the inference process.

In addition, other studies focus on a realtime implementation of the stereo disparity estimation. Khamis *et al.* [21] proposed the first realtime end-to-end network (StereoNet) with 1/16 original resolution to regress the disparity map and a post-processing step to refine the coast disparity by dilated convolution. Tonioni *et al.* [50] proposed an unsupervised, lightweight and effective continuous online network (MADNet) to reduce computational cost. Further, Wang *et al.* [1] presented the AnyNet, which not only obtained a better performance but also used less parameters (about 1/10 parameters) than StereoNet [21].

Although recent approaches have made significant success, there's still a long way to make stereo disparity networks even more lightweight and realtime. Following previous research, we would like to expand the research using knowledge distillation to further improve the performance of stereo disparity estimation. The knowledge distillation has shown great performance in various fields such as face recognition [51], object detection [52], speech recognition [53] and

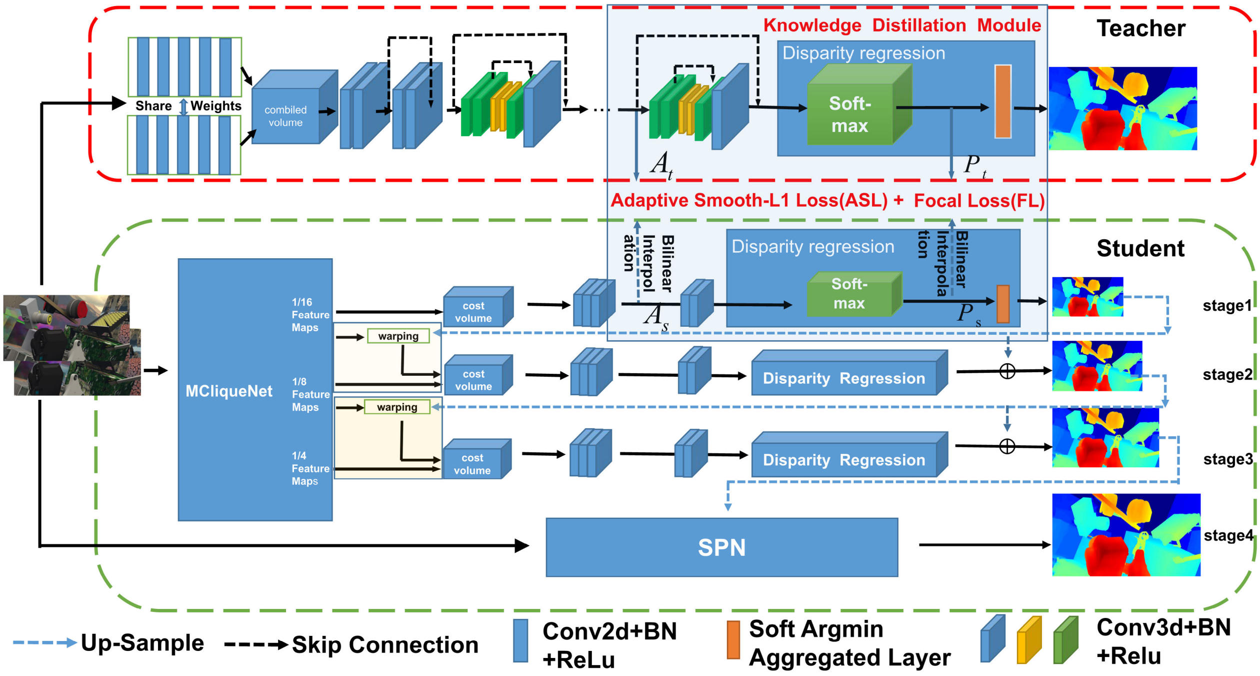


FIGURE 2. The whole framework and data pipline proposed in this paper. It consists of a teacher network [20] (red dash part) and a student network (green dash part), where MCliqueNet was shown in Figure 3. The cost volume was mimicked after the 3D convolution filtering and the *softmax* operation (blue dash part). Distillation loss module will be introduced in Section IV-D3 in detail.

pose regressor [54]. However, most studies using knowledge distillation have been well applied in classification task but not for solving a regression problem like disparity estimation. Therefore, in this paper, we are committed to investigate how knowledge distillation can be used to improve the performance of stereo disparity estimation.

III. PROPOSED METHODS

Based on the aforementioned problem, we proposed a novel stereo disparity estimation method with less parameters to learn the high accuracy yet fast CNN architecture. An overview of the architecture is shown in Figure 2, which consists of a teacher and a student network. In this paper, the teacher network is a big network. We use the state-of-the-art network–GwcNet [20], which has a mount of stacked 3D convolutions with approximate 4.48M parameters. The student network is a lightweight cascaded network with 0.042M parameters, which is two order less than the teacher network. For the teacher network, more details can be found in GwcNet [20]. In this paper, we present only the components of the student network.

A. MULTI-SCALE CliqueNet (MCliqueNet) FEATURE EXTRACTOR

For the stereo disparity estimation task, we believe that more refined features can make the probability of the mismatch lower. AlexNet [55], VGG [56], ResNet [57], DenseNet [58] and CliqueNet [59] play an important role in the development of feature extraction networks. However, in our study, how to design a lightweight and efficient feature extractor is crucial for stereo disparity estimation. Considering the number of parameters and the computational complexity, the CliqueNet architecture is a good choice. The CliqueBlock of CliqueNet [59] can help to ease the training difficulties and utilize parameters more efficiently and achieves more refined features with smaller parameters. Therefore, the CliqueBlock is chosen as the base unit in our feature extractor. In order to further reduce the computational load of the entire system, we designed a multi-scale CliqueNet, instead of using the traditional CliqueNet directly. The multi-scale CliqueNet converts a unique high-resolution image feature extraction task into a multi-scale feature extraction task. CliqueBlock [59] after downsampling cascade can reduce the images directly extracted at original resolution, which can reduce the resource consumption of high-resolution image feature extraction only.

The proposed MCliqueNet architecture mainly consists of three CliqueBlocks [59] and two adaptive channel attention transition modules. The overview of network can be seen in Figure 3. Considering not adding an additional computational burden, we only perform one cycle for feature refinement with CliqueBlock. Before the first CliqueBlock architecture, we extract convolutional features with the kernel size of 7 × 7. Figure 3 illustrates the structure of MCliqueNet in detail, which is shared by left and right images. Each CliqueBlock will be aggregated with previous block after

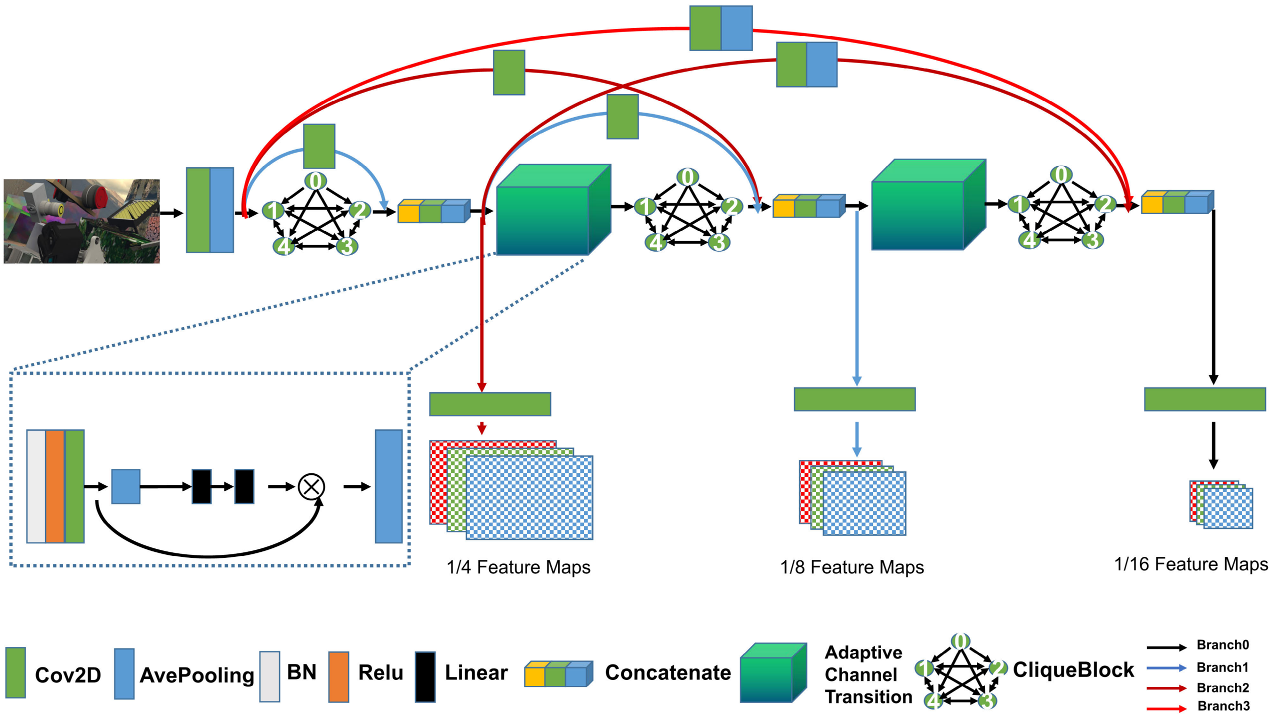
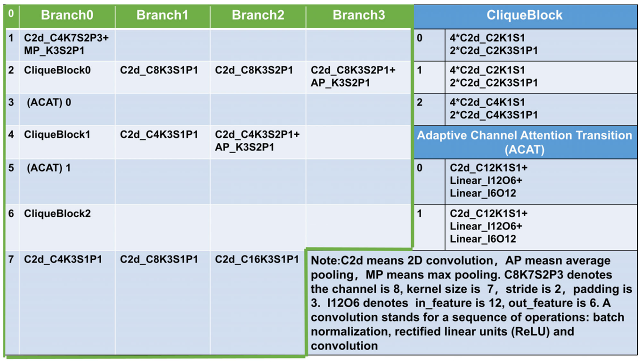


FIGURE 3. The structure of the proposed MCliqueNet. It mainly consists of three CliqueBlocks [59] and two adaptive channel attention transition modules. The specific network parameters are given in detail in Figure 4.



0	Branch0	Branch1	Branch2	Branch3
1	C2d_C4K7S2P3+ MP_K3S2P1			
2	CliqueBlock0	C2d_C8K3S1P1	C2d_C8K3S2P1	C2d_C8K3S2P1+ AP_K3S2P1
3	(ACAT) 0			
4	CliqueBlock1	C2d_C4K3S1P1	C2d_C4K3S2P1+ AP_K3S2P1	
5	(ACAT) 1			
6	CliqueBlock2			
7	C2d_C4K3S1P1	C2d_C8K3S1P1	C2d_C16K3S1P1	

CliqueBlock	
0	4*C2d_C2K1S1 2*C2d_C2K3S1P1
1	4*C2d_C2K1S1 2*C2d_C2K3S1P1
2	4*C2d_C4K1S1 2*C2d_C4K3S1P1

Adaptive Channel Attention Transition (ACAT)	
0	C2d_C12K1S1+ Linear_I12O6+ Linear_I6O12
1	C2d_C12K1S1+ Linear_I12O6+ Linear_I6O12

Note:C2d means 2D convolution, AP measn average pooling, MP means max pooling. C8K7S2P3 denotes the channel is 8, kernel size is 7, stride is 2, padding is 3. I12O6 denotes in_feature is 12, out_feature is 6. A convolution stands for a sequence of operations: batch normalization, rectified linear units (ReLU) and convolution

FIGURE 4. The network parameters of the proposed MCliqueNet. There are four branches in total, and the parameters on each branch are shown here in detail.

3 × 3 convolution and average pooling, which aims to get more contextual convolutional features and global features in terms of both spacial and channel information. Next, the aggregated features are fed into next CliqueBlock by adaptive channel attention scheme. The purpose of this module is to downsample high-resolution features while better preserving the features extracted at high resolution. After the adaptive channel attention transition module, the size of the feature map will be rescaled to half of the original size, but it does not change the number of channels. The parameter details of the proposed network can be obtained in Figure 4, which is divided into four branches to show the modules and each branch and distinguished by different colored pipelines.

As mentioned above, in order to improve the robustness of the features and reduce the probability of mismatches in the stereo disparity matching process, we decided to merge multi-scale features. Therefore, we utilize the output of three concatenation layers to aggregate different levels of feature maps, which can produce feature maps in different resolution (1/16x, 1/8x, 1/4x).

B. THE CONSTRUCTION OF COST VOLUME

Once we have the multi-scale feature maps, we need to build a cost volume. The proposed method generates a 5-dimensional cost volume to construct the relationship between a real 3D world and a 2D image in different levels. The cost volume represents the matching cost between the left and right features from 0 to maximum disparity for every pixel. In order to trade off complexity and accuracy, we combined the results after subtraction between the left and right features with L1-norm to construct the cost volume instead of using the group-wise cost volume [20] directly, which also can be seen in Figure 5. We marked the corresponding features that need to be subtracted with the same color block, such as blue, red, green, and etc. and the part without color is filled with zeros.

As we know, the most complex part of the deep learning model in stereo disparity estimation is to refine cost volume. Generally, we need to construct a 5-dimensional cost volume of $B \times C \times M \times H \times W$, where M denotes the maximum disparity, and the typical $M = 192$, H, W respectively represent the resolution of the input image, B is the batchsize, C is the cost volume channel (here $C = 1$). If the 5-dimensional

TABLE 1. List of aggregational components at different resolutions.

Resolution	Components				
1/16	BRCi1o16k3s1p1	BRCi16o16k3s1p1	BRCi16o16k3s1p1	BRCi16o16k3s1p1	BRCi16o1k3s1p1
1/8	BRCi1o4k3s1p1	BRCi4o4k3s1p1	BRCi4o4k3s1p1	BRCi4o4k3s1p1	BRCi4o1k3s1p1
1/4	BRCi1o4k3s1p1	BRCi4o4k3s1p1	BRCi4o4k3s1p1	BRCi4o4k3s1p1	BRCi4o1k3s1p1

where B denotes BatchNormal3D, R denotes ReLU, C denotes Conv3D. i means input channel, o means output channel, k means kernel, s means stride, p means padding. BRCi1o16k3s1p1 means a module consists of BatchNormal, ReLU, Conv3D, and the parameters of Conv3D are input channel =1, output channel= 16, kernel size= 3, stride= 1, and padding= 1.

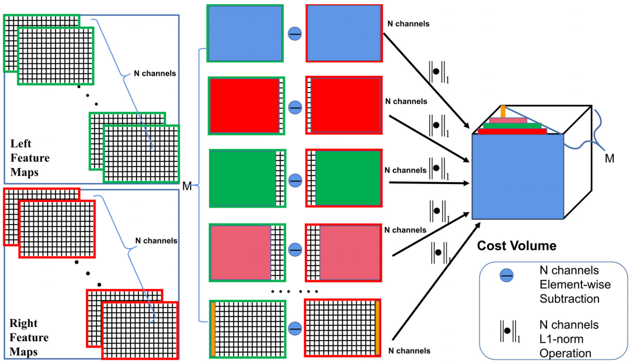

N channels
Left Feature Maps
Right Feature Maps
M
Cost Volume
N channels Element-wise Subtraction
N channels L1-norm Operation

FIGURE 5. The process of building cost volume. According to the obtained feature map, we combined the results after subtraction between the left and right features with L1-norm to construct the cost volume, the part without color is filled with zeros.

cost volume is sent directly to the cascade 3D convolution, the computational cost is very large. Therefore, in our task, in order to reduce the computational burden of 3D convolution, we built only 1/16 resolution cost volume where $M = 192/16$.

For the 1/4 and 1/8 feature maps, in order to allow the 3D convolution only to learn an offset based on the previous stage, we simply construct a residual cost volume by warping the features in the same space. In particular, the corresponding offset is $-2, -1, 0, 1, 2$ ($M = 5$), which can reduce the computation significantly.

C. DISPARITY AGGREGATION AND REGRESSION

For disparity aggregation, although utilizing a lot of 3D convolutions is a good choice to merge geometry and context information [17], the network will be at a extremely heavy computational burden. Therefore, in this work, we used a few 3D convolutions to aggregate disparity (The number of 3D convolution is 5).

In order to reduce the computational load of 3D convolution, the number of channels of 3D convolution is different at different resolutions. Table 1 shows in details.

For the disparity regression, traditional methods use a WTA (winner to take all) strategy. However, WTA is not differentiable in a fully end-to-end training. Kendall *et al.* [17] proposed a differential *soft argmin* method, which can be written as follows:

$$\hat{d} = \sum_{d=0}^{D_{max}} d \times \sigma(-A) \tag{1}$$

where d and A denote the disparity level and the filtered cost volume after 3D convolution respectively. $\hat{d}$ represents the estimated disparity. The $\sigma(\cdot)$ denotes the *softmax* operation.

After this method was proposed, it has been widely used in the stereo disparity estimation. Therefore, a *soft argmin* method is also used in this work.

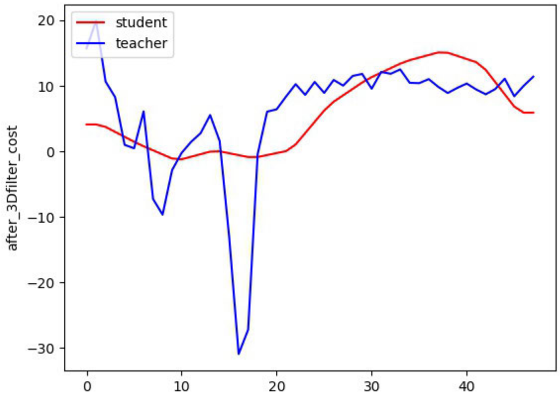


FIGURE 6. The matching cost. The horizontal axis represents the disparity level, and the vertical axis represents the estimated cost of the corresponding disparity, with a smaller cost indicating a higher matching.

D. KNOWLEDGE DISTILLATION OF COST VOLUME

In this part, we will state how to improve the performance of small network by knowledge distillation scheme. The high accuracy and large networks is called teacher networks, while the low accuracy and small is called student networks. In stereo disparity estimation pipeline, cost aggregation is a critical step, which affects the accuracy and efficiency. Most previous works take advantage a lot of 3D convolutions to obtain accurate disparity, resulting in a high computational cost and requiring a mount of memory resource. In order to fully exploit the performance of small networks, we have analyzed and compared the cost volume between the large network and the small network. We found that the cost volume of teacher network after 3D convolution filtering has extremely low matching cost in narrow range at corresponding disparity, but the student networks are in a wide range. The results are illustrated in Figure 6. Thus, we believe that if we could transfer the characteristic of the teacher network to the student network, then the performance of the student network could be potentially improved without any cost. In order to make a student network mimic the cost volume of a teacher network, mean square error (L2 loss) is used to minimize the distance between cost volumes of the teacher network and the student network. However, L2 Loss is not robust which can be easily affected by outliers. In this paper, we proposed to utilize a $Smooth_{L1}$ loss for calculating the similarity of cost volumes between teacher and student networks. The $Smooth_{L1}$ loss can be written as follows:

$$Smooth_{L1}(x) = \begin{cases} 0.5x^2, & \text{if } |x| < 1 \\ |x| - 0.5, & \text{otherwise} \end{cases} \tag{2}$$

$Smooth_{L1}$ can not only reduce the influence of outliers, but also have the advantage of L2 Loss, which changes the gradient as the loss changes. The first part of cost volume loss of distillation knowledge can be represented as follows:

$$L_{cvs} = \frac{1}{HWD}\sum_{i=0}^{HW}\sum_{j=0}^{D} Smooth_{L1}(a_t^{ij} - a_s^{ij}) \tag{3}$$

where H, W, D denote the height,width and disparity of cost volume. $a_t^{ij} \in A_t$,,$a_s^{ij} \in A_s$, A_s and A_t denote cost volume of the teacher network and the student network after 3D convolution filtering respectively, which can be seen in Figure 2.

In this paper, although we use the SOTA algorithm GwcNet [20] as teacher network, the pre-trained teacher network may not be accurate in inference processing for some cases, which may guide student worse. If we simply allow student networks to learn teacher networks without choice, their performance will be poorer on some cases. To resolve aforementioned problem, we proposed an adaptive $Smooth_{L1}$ (ASL) Loss, which can adjust the contribution of the teacher network by using the error between the teacher network and the ground truth. Therefore, misdirection can be filtered during knowledge distillation by adaptively adjusting K_j. The function can be rewritten as follows:

$$L_{acvs} = K_j L_{cvs} \tag{4}$$

where K_j is an adaptive weight, it can be written as follows:

$$K_j = \left(1 - \frac{EPE(D_T^j, D_{GT}^j) - min(E)}{max(E) - min(E)}\right) \tag{5}$$

where D_T^j denotes the jth prediction of teacher, D_{GT}^j denotes the ground truth. $E := \{EPE(D_T^k, D_{GT}^k)|k = 1, 2, \ldots, N\}$, N is the total number of training datasets, EPE means the end-point-error. Therefore, as long as we first infer the teacher network on the training datasets, we can obtain the max(E) and min(E). Please note that the K_j decreases as $EPE(D_T^j, D_{GT}^j)$ increases. It can also be interpreted in a different way: when the teacher network output has a large error, we should adjust its teaching contribution (K_j should be decreased), which aims to ensure the accurate knowledge of the teacher network.

After 3D convolution filtering, the cost volume will be sent into the *softmax* operation, which yields the distribution of probability of disparity from 0 to max disparity. From Figure 7, we can see that the distribution of the teacher network is unimodel and concentrate, but the student is bimomal even muti-peak. Therefore, the student network is ambiguous. After Equation 1, it will produce errors at the disparity map. In order to tackle this problem, we make the student network to learn the distribution of the teacher network using a Cross Entropy (CE) Loss. Furthermore, the distribution of disparity has more negative samples than the positive samples (the value of distribution of disparity is zero in most disparity level). In order to reduce the influence of negative samples, inspired by Focal Loss(FL) [31], we modified the loss function in order to apply knowledge distillation to regress disparity. It can be written as follows:

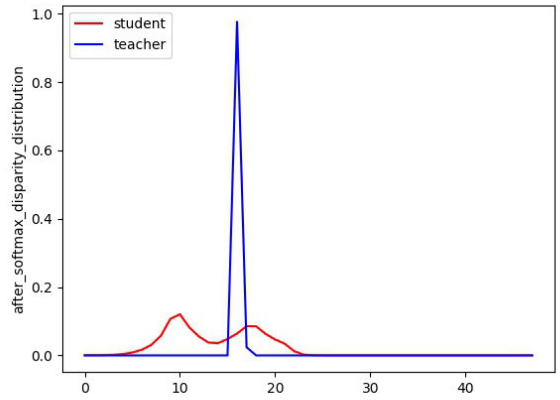


FIGURE 7. **The probability of distribution. The horizontal axis represents the disparity level and the vertical axis the estimated probability of corresponding disparity.**

$$L_{cvdf} = \frac{1}{HW}\sum_{i=0}^{HW}\Big(\sum_{d=0}^{D_{max}}(1 - p_s^i(d))^{-\gamma}(-p_t^i(d) \cdot log(p_t^i(d)))\Big) \tag{6}$$

where H and W denote the height and width of cost volume, $p_s^i(d) \in P_s$, $p_t^i(d) \in P_t$. P_s and P_t denote the distribution of the student network and the teacher network respectively, which can be seen in Figure 2.

In this paper, since the cost volume of teacher network is on a 1/4 scale but the cost volume of student network is on a 1/16 scale, we should up-sample(x4) the cost volume of student network with bilinear interpolation as the same size of the teacher network (the teacher is $B \times C \times D/4 \times H/4 \times W/4$, while the cost volume of the student network is $B \times C \times D/16 \times H/16 \times W/16$). For example, we should un-sample the A_s to fit the size of A_t. Please keep in mind that we only up-sample the cost volume of the student once when calculated the loss between teacher and student, we did not further down-sample the up-sampled cost volume to aggregate disparity, which also can be seen in Figure 2. The reason we don't down-sample the cost volume of the teacher is that down-sampling may result in the loss of teacher knowledge, which will affect the learning process of student.

E. SPATIAL PROPAGATION NETWORK (SPN)

In the last phase, we also used the SPN network [60] to further improve performance which can refine our disparity predictions. The principle of SPN is to use the local similarity of the left image to refine the disparity map. Please refer to the original article to get more details.

F. LOSS FUNCTION

We train our network in a fully end-to-end supervised way, which means the network directly estimated disparity image through only a pair of stereo images. The proposed total loss is written as follows:

$$L_{total} = \lambda_{d1} L_{acvs} + \lambda_{d2} L_{cvdf} + L_{dis} \tag{7}$$

where λ_{d1} and λ_{d2} are the weight of L_{acvs} and L_{cvdf} respectively. L_{dis} is the $Smooth_{L1}$ loss between the estimated disparity and ground-truth, which can be written as follows:

$$L_{dis} = \sum_{i=0}^{3} \sum_{0}^{H_i W_i} \frac{\lambda_i}{H_i W_i} Smooth_{L1}(d_i^* - \hat{d}_i) \tag{8}$$

where d^* is ground truth. $\hat{d}_i$ denotes the estimated disparity of ith stage. λ_i is the weight of ground truth and estimated disparity of the ith stage. H_i and W_i denote the height and width at different stages. The whole process can be clearly understood in Figure 2.

IV. EXPERIMENTS AND RESULTS

In this section, we thoroughly evaluated the performance of our proposed network architecture—Compact StereoNet on Scence Flow [41], KITTI2012 [61] and KITTI2015 [62] dataset at different settings. We reproduced Anynet and compared the achieved results in our study. For a fair comparison, the performance of other methods have been achieved using the same dataset and the same validation process. For those metrics that have not been reported in previous works, we have re-implemented the method in this study.

A. IMPLEMENTS DETAILS

The whole networks including a teacher network and a student network were implemented using PyTorch 0.4. Firstly, we pre-trained a teacher network—GwcNet [20] with 15 epochs on Scene Flow dataset [41]. After that, we jointly trained the student network and the teacher network. The batch size was set to 4 and Adam [63] was used for optimization with $\beta_1 = 0.9, \beta_2 = 0.999$. The training process was performed on a Nvidia GTX 1080 and required about 2 hours for an epoch. The Scene Flow dataset was used for training and the process was stopped after 20 epochs. The initial learning rate was set to 1e-3, and was decreased to 0.3 of the previous value every 4 epochs. Especially, we set $\lambda_{d1} = 0.2, \lambda_{d2} = 0.3, \lambda_0 = 0.5, \lambda_1 = 0.5, \lambda_2 = 1, \lambda_3 = 1$.

After training and testing on the Scene Flow dataset, we finetuned the pre-trained model on the KITTI2012 and KITTI2015 dataset for 300 epochs respectively. The initial learning rate was set to 1e-4. After 200 epochs, the learning rate was changed to 1/10 of the original value. As in the AnyNet [1] architecture, before 120*th* epoch, the proposed method was trained without a SPN [60] module.

B. EVALUATION METRICS

For a fair comparison with state-of-the-art methods, we used end-point-error (EPE) and three-pixel-error as evaluate metrics on the Scene Flow dataset. The EPE can be written as (9), which means the average pixel-wise disparity error. The three-pixel-error (T3) that defines as the absolute of pixel error more than 3 pixels, which can be written as (11). Besides, one-pixel-error (T1) and two-pixels-error (T2) were also reported additionally to fully evaluate the performance of the proposed network.

For the KITTI dataset, in addition to the previous evaluation metrics, we used the D1-all metric to evaluate, which defines as the pixels whose disparity errors are the larger of 3 pixels or 5% real disparity.

$$EPE = \frac{1}{N} \sum_{0}^{N} \left\| GT[mask] - \hat{D}[mask] \right\|_1, \tag{9}$$

where mask $\in \{0, 1\}^{HW}$. H, W represents the height and width of image. The mask can be calculated as (10). N denotes the number of 1.

$$m = \begin{cases} 1, & \text{if } 0 < d < 192 \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

where m $\in$ mask, d denotes the disparity of GT.

$$T3 = \frac{1}{N} \sum_{0}^{N} \left| GT[mask] - \hat{D}[mask] \right|, \tag{11}$$

where mask $\in \{0, 1\}^{HW}$, However, the N is the number of mask=1 and $|GT[mask] - \hat{D}[mask]|$ is greater than 3. A similar definition applies to T1, T2 and D1-all, but the calculation of N is changed.

C. COMPARISON WITH OTHER METHODS

In this section, the quantitative results on the Scene Flow dataset, the KITTI2012 dataset and the KITTI2015 dataset are shown in Table 2, 3 and 4 respectively. All compared methods were implemented on the same platform using the same input on a single Nvidia GTX 1080 with the setting of batch_size = 1.

1) SCENE FLOW

It is a big synthetic dataset, which contains 4370 group testing data. We tested all of the 4370 group datasets, the quantitative results can be seen in Table 2.

As shown in the Table 2, we compared the performance of the proposed method with StereoNet [21] which has 0.31MB parameters, but it is about five times as large as the number of parameters in our proposed network. The EPE value of StereoNet is 3.558, while our method achieves an EPE of 2.771. The error rate is reduced by approximate 22.77%. At the same time, it costs more time than our network. Since other T1, T2, and T3 metric are not reported in the original article, we do not report them either.

TABLE 2. Quantitative results on the scene flow testing dataset. The result of StereoNet is reported by author with 16x, and unrefinement [21].

Model	Params(MB)	Feature Extractor	Distillation Loss			Evaluate Metrics				Time(s)
			SL1	ASL	FL	EPE(px)	T1(%)	T2(%)	T3(%)	
GwcNet (teacher CVPR2019) [20]	4.48	-	-	-	-	0.765	8.07	4.47	3.30	-
GC-Net (ICCV2017) [17]	3.5	-	-	-	-	2.51	-	-	-	0.900
DESNet (CVPR2018) [64]	42.76	-	-	-	-	2.81	-	-	-	0.060
DenseMapNet (ICRA 2018) [65]	0.29	-	-	-	-	5.07	-	-	-	0.020
StereoNet (ECCV2018) [21]	0.31	-	-	-	-	3.588	-	-	-	0.052
AnyNet (ICRA2019) [1]	0.043	U-Net	-	-	-	3.403	33.70	21.90	16.90	0.013
AnyNet v1	0.043	U-Net	✓	×	✓	3.334	33.19	21.65	16.67	0.013
AnyNet v2	0.043	U-Net	×	✓	✓	3.320	32.52	21.42	16.57	0.013
Compact StereoNet v1	0.042	MCliqueNet	×	×	×	2.952	30.60	20.10	15.40	0.012
Compact StereoNet v2	0.042	MCliqueNet	✓	×	✓	2.828	29.66	19.26	14.76	0.012
Compact StereoNet v3	0.042	MCliqueNet	×	✓	✓	**2.771**	**29.21**	**19.00**	**14.57**	0.012

T1, T2 and T3 denote the absolute error pixel larger than 1 pixel, 2 pixels and 3 pixels. EPE means end-point-error. FL means Focal Loss with $\gamma = 2$. SL1 means $smooth_{L1}$ Loss . ASL means adaptive SL1 Loss.

TABLE 3. Quantitative results on the KITTI2012 dataset.

Model	Evaluate Metrics					Time(s)
	EPE(px)	D1-all	T1(%)	T2(%)	T3(%)	
AnyNet (ICRA2019) [1]	1.198±0.18	6.30±0.58	34.80±2.5	12.57±1.4	6.2±0.6	0.011
Compact StereoNet v1	1.071±0.12	5.52±0.55	22.43±1.99	9.64±1.22	5.84±0.55	0.009
Compact StereoNet v3	**1.043±0.10**	**5.32±0.55**	**21.89±1.9**	**9.23±1.1**	**5.58±0.48**	0.009

The Compact StereoNet v1 means that the knowledge distillation was not used while the Compact StereoNet v3 means that the knowledge distillation with ASL loss and FL was used, which also can be seen in Table 2. EPE means end-point-error. D1-all define as the pixels whose disparity errors are the larger than 3 pixel or 5% real disparity including forward region and background region. Please note that time was calculated on a Nvidia GTX 1080 with batch_size=1, and the size of testing image is 3 × 1232 × 368.

TABLE 4. Quantitative results the on KITTI2015 dataset.

Model	Evaluate Metrics					Time(s)
	EPE(px)	D1-al1(%)	T1(%)	T2(%)	T3(%)	
AnyNet (ICRA2019) [1]	1.203±0.14	6.34±0.6	34.64±2.7	12.58±1.9	6.37±0.9	0.011
CompactStereoNet v1	1.047±0.11	5.28±0.55	22.43±2.2	9.64±1.1	5.84±0.74	0.009
CompactStereoNet v3	**0.999±0.09**	**4.71±0.55**	**21.89±1.94**	**9.23±0.99**	**5.58±0.50**	0.009

The Compact StereoNet v1 means that the knowledge distillation was not used while the Compact StereoNet v3 means that the knowledge distillation with ASL loss and FL was used, which also can be seen in Table 2. EPE means end-point-error. D1-all define as the pixels whose disparity errors are the larger than 3 pixel or 5% real disparity including forward region and background region. Please note that time was calculated on a Nvidia GTX 1080 with batch_size=1, and the size of testing image is 3 × 1232 × 368.

As for AnyNet [1], it uses U-Net [66] as the feature extractor. we use it as our baseline. Although it has been already a state-of-the-art work in lightweight network, our metrics far surpass it as well. We implemented its work and its metrics are also close to those reported in the paper. We have compared its performance and our methods at all of stages, including stage 1, 2, 3, and 4. However, in this section, we only show the comparison at 4th stage. The performance of the remaining three stages will be shown in section IV-D3. Actually, all of stages are improved significantly. Especially, the improvement is the most significantly at the first stage. As we can see from Table 2, although the number of parameters and the time consuming of our model are almost same even slightly lower with AnyNet, the EPE value dropped from 3.403 to 2.771, which is a significantly improvement by our proposed method. Other indicators such as T1, T2, T3 are also significant improved.

Comparing the state-of-the-art lightweight CNN networks including the StereoNet [21] and the AnyNet [1] for stereo disparity estimation, our method outperforms them in all evaluation metrics under similar computational complexity. The qualitative results between AnyNet and the proposed Compact StereoNet are shown in Figure 8. For the qualitative of StereoNet, we cannot reproduce its results. Therefore, we just to report the quantitative results using original paper.

2) KITTI2012

The dataset have 194 groups training image. For a fair comparison, we performed five folds cross validations. We calculated the mean and variance of the five fold cross validations.

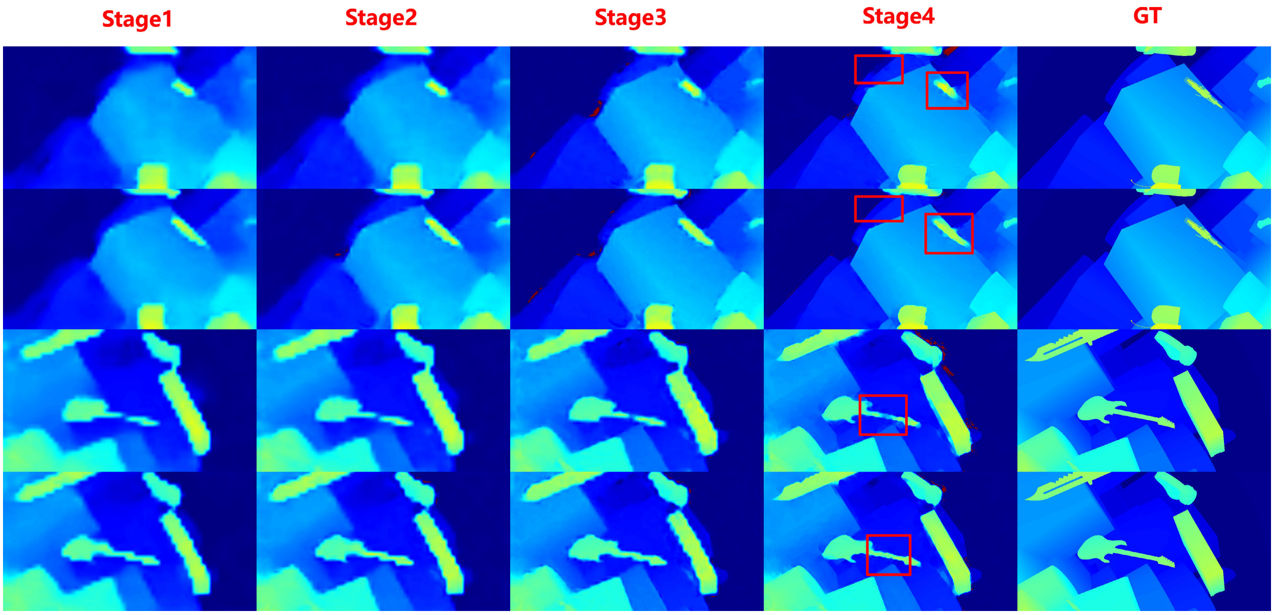


FIGURE 8. Qualitative results on the scene flow dataset. The red box region can easily distinguish the difference. 1*th* and 3*th* rows are the results using AnyNet, while the 2*th* and 4*th* rows are the results of the proposed Compact StereoNet v3. We can find that the results of Anynet are bad for some edge processing.

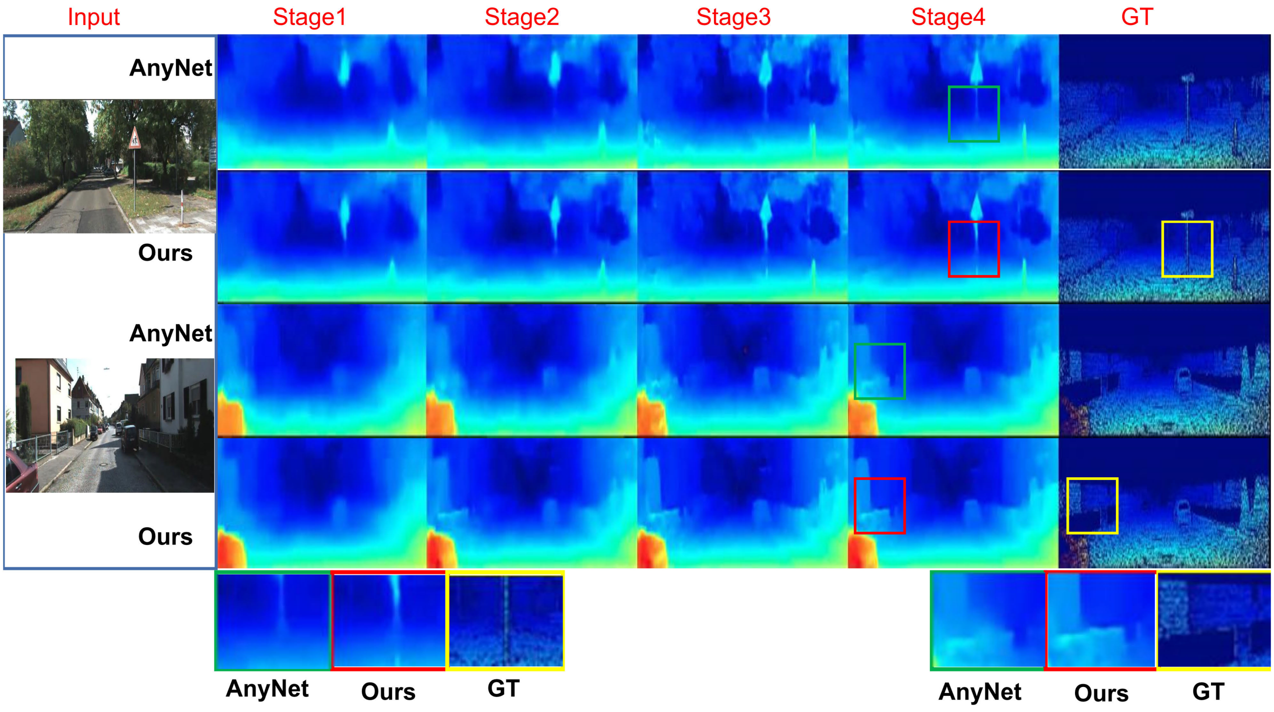


FIGURE 9. Quantitative results on the KITTI2012 dataset. The color box areas can easily distinguish the difference. 1*th* and 3*th* rows are the results using AnyNet, while the 2*th* and 4*th* rows are the results of the proposed Compact StereoNet v3. The Compact StereoNet v3 means that the knowledge distillation with ASL loss and FL was used, which also can be seen in Table 2.

The quantitative results are shown in Table 3. As shown in Table 3, we can see that the proposed compact StereoNet v1 without using knowledge distillation scheme can decrease the EPE value about 10.6% over the AnyNet. Moreover, if we use the compact StereoNet v1 with knowledge distillation can be further decreased the EPE metric about 2.61%. The qualitative results are shown in Figure 9 between AnyNet and the proposed Compact StereoNet v3.

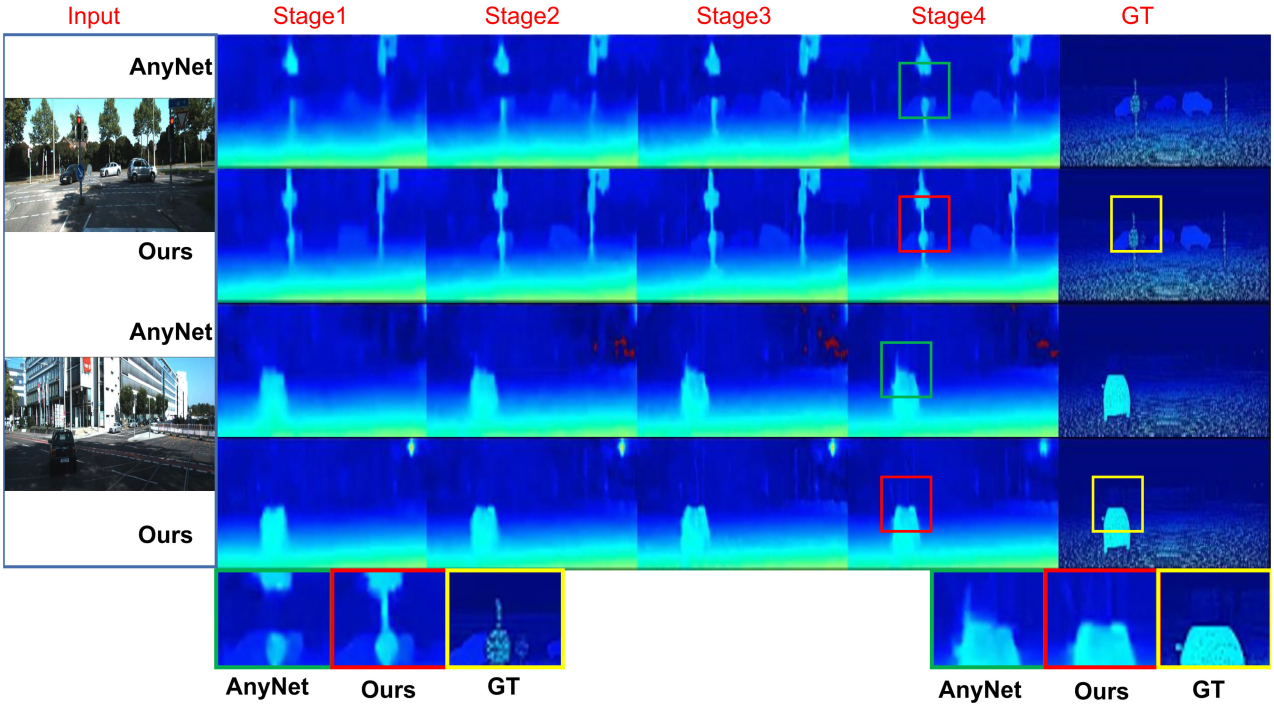


FIGURE 10. Quantitative results on the KITTI2015. The color box areas can easily distinguish the difference. 1*th* and 3*th* rows are the results using AnyNet, while the 2*th* and 4*th* rows are the results of the proposed Compact StereoNet v3. The Compact StereoNet v3 means that the knowledge distillation with ASL loss and FL was used, which also can be seen in Table 2.

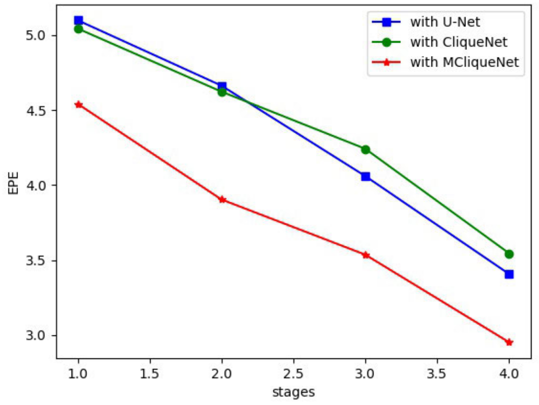


FIGURE 11. The comparison of results using different network for feature extraction on same disparity estimation network (AnyNet) about SceneFlow dataset. We can see that the performance of disparity estimation can be significantly improved with MCliqueNet.

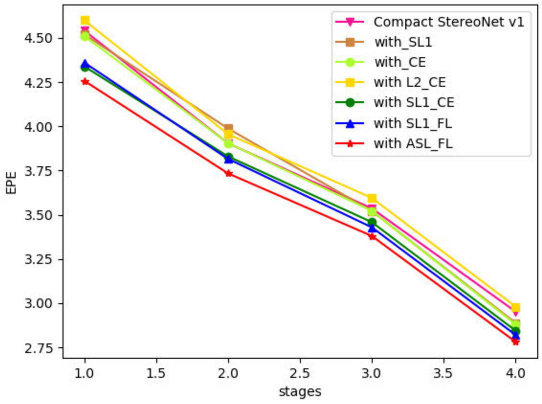


FIGURE 12. The results of the proposed compact StereoNet with different losses on the Scene Flow dataset. The compact StereoNet v1 means that the knowledge distillation was not used, which also can be seen in Table 2. L2 denotes that the distillation loss function is L2 Loss, SL1 means $Smooth_{L1}$, ASL means adaptive $Smooth_{L1}$, CE means cross entropy loss. FL means focal loss with $\gamma = 2$.

3) KITTI2015

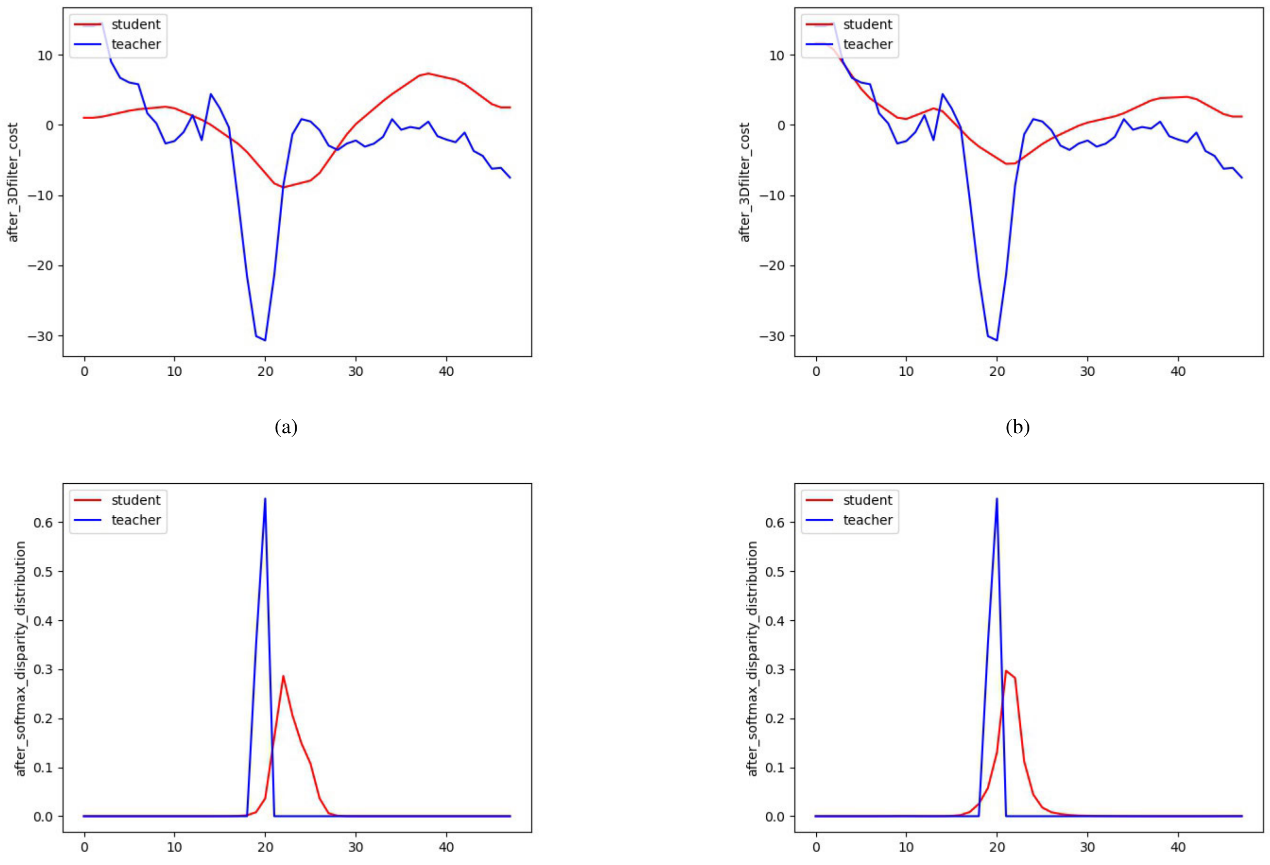
The dataset have 200 groups training image. We also performed five folds cross validation. As shown in Table 4, our method surpasses AnyNet in all evaluation metrics. The qualitative results are shown in Figure 10 between AnyNet and the proposed Compact StereoNet v3. The Compact StereoNet v3 means using knowledge distillation with ASL loss and FL, which also can be seen in Table 2.

D. ABLATION STUDY

In this section, we investigate the effect of each module on stereo disparity accuracy to further validate our proposed approach.

1) FEATURE EXTRACTOR

We first evaluated our proposed MCliqueNet with extracting feature. We use U-Net [66] as our baseline. As shown



(a)

(b)

(c)

(d)

FIGURE 13. The matching cost and probability of distribution at *ith* pixel before and after distillation.(a), (b) are the matching cost at *ith* pixel before distillation and after distillation separately. (c), (d) are the distribution of probability at *ith* pixel before distillation and after distillation separately.

in Figure 11, the performance is not improved by using traditional CliqueNet at all stages even the number of parameters is larger than that of U-Net (the disparity estimation network with traditional CliqueNet is 0.044MB, while the AnyNet is 0.043MB). On the contrary, it makes the performance worse in stage 3 and stage 4. The results are slightly improved at stage 1 and 2. The reason for this is that the deep features are not well fused with shallow features. Overall, the accuracy of disparity can be improved about 13.2% by using our proposed MCliqueNet network which almost have a similar number of parameters and computational cost as that of AnyNet. It demonstrates that the fusion of shallow and deep features is beneficial for stereo matching. Moreover, from Table 2, we can see that a network using U-Net as a feature extractor, even with the knowledge distillation scheme, performs far worse than one using our proposed MCliqueNet. Meanwhile, if the original AnyNet is distilled directly, the EPE decreased 2.44% and other evaluation metrics are slightly improved. If we use the proposed MCliqueNet as the extracting feature module, the EPE has been significantly decreased at 6.13%.

2) DIFFERENT DISTILLING LOSSES

To evaluate the influence of different distilling losses, we trained the same CNN architecture (Compact StereoNet v1) with different losses. The results are shown in Figure 12. First, we experimented with a single instruction i.e., using only SL1 or CE loss, and we can achieve a slight improvement. If we combine SL1 with CE loss, the performance will be significantly improvement. However, the performance are degraded with L2 Loss as it is sensitive to outliers. If we use ASL loss, the accuracy of disparity will be further improved. Obviously, the combination of the ASL loss and the FL is the best choice among all losses.

3) EFFECTS OF DISTILLATION

We have also analyzed the distillation effect of the cost volume. The distilling effects are shown in Figure 13. As the shown in Figure 13, Comparing 13 (a) and 13 (b) or 13 (c) and 13 (d), the cost volume after distillation between student and teacher network is more similar. Therefore, the teacher network can correct the student network by using knowledge distillation scheme.

V. CONCLUSION

In this paper, we proposed a compact convolutional neural architecture–MCliqueNet, which is suitable for stereo disparity estimation to extract features. Furthermore, we have proposed a lightweight network using knowledge distillation to significantly improve its performance. Finally, we present a novel adaptive $Smooth_{L1}$ (ASL) loss for calculating similarity between the cost volumes of the teacher network and the student network. We have demonstrated the effectiveness of our proposed method through extensive experiments and ablation studies. Experimental results show that our method achieves competitive performance on the challenging Scene Flow and KITTI benchmarks while maintaining a very fast running time.

In future, we will try different distillation methods to improve the performance of small student networks for stereo disparity estimation.

REFERENCES

[1] Y. Wang, Z. Lai, G. Huang, B. H. Wang, L. van der Maaten, M. Campbell, and K. Q. Weinberger, "Anytime stereo image depth estimation on mobile devices," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 5893–5900.

[2] K. Ma, H. Zhou, J. Li, and H. Liu, "Design of binocular stereo vision system with parallel optical axesand image 3D reconstruction," in *Proc. China-Qatar Int. Workshop Artif. Intell. Appl. Intell. Manuf. (AIAIM)*, Jan. 2019, pp. 59–62.

[3] R. Fan, X. Ai, and N. Dahnoun, "Road surface 3D reconstruction based on dense subpixel disparity map estimation," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 3025–3035, Jun. 2018.

[4] H. Abu Alhaija, S. K. Mustikovela, L. Mescheder, A. Geiger, and C. Rother, "Augmented reality meets computer vision: Efficient data generation for urban driving scenes," *Int. J. Comput. Vis.*, vol. 126, no. 9, pp. 961–972, Sep. 2018.

[5] N. Zenati and N. Zerhouni, "Dense stereo matching with application to augmented reality," in *Proc. IEEE Int. Conf. Signal Process. Commun.*, Nov. 2007, pp. 1503–1506.

[6] F.-Y. Hsiao and P.-Y. Lee, "Autonomous indoor passageway finding using 3D scene reconstruction with stereo vision," in *Proc. Comput. Conf.*, Jul. 2017, pp. 279–285.

[7] Z. Zhang, X. Ai, N. Canagarajah, and N. Dahnoun, "Local stereo disparity estimation with novel cost aggregation for sub-pixel accuracy improvement in automotive applications," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2012, pp. 99–104.

[8] J. Huang, S. Tang, Q. Liu, and M. Tong, "Stereo matching algorithm for autonomous positioning of underground mine robots," in *Proc. Int. Conf. Robots Intell. Syst. (ICRIS)*, May 2018, pp. 40–43.

[9] X. Shao, Y. Yang, and W. Wang, "Obstacle crossing with stereo vision for a quadruped robot," in *Proc. IEEE Int. Conf. Mechatronics Autom.*, Aug. 2012, pp. 1738–1743.

[10] G. Zhai, W. Zhang, W. Hu, and Z. Ji, "Coal mine rescue robots based on binocular vision: A review of the state of the art," *IEEE Access*, vol. 8, pp. 130561–130575, 2020.

[11] H. Hirschmuller, "Accurate and efficient stereo processing by semi-global matching and mutual information," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 807–814.

[12] J. Sun, Y. Li, S. Bing Kang, and H.-Y. Shum, "Symmetric stereo matching for occlusion handling," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 399–406.

[13] O. Veksler, "Fast variable window for stereo correspondence using integral images," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1, Jun. 2003, pp. 556–561.

[14] K.-J. Yoon and I. So Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 650–656, Apr. 2006.

[15] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.*, vol. 47, nos. 1–3, pp. 7–42, Apr. 2002.

[16] J. Žbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches," *J. Mach. Learn. Res.*, vol. 17, nos. 1–32, p. 2, 2016.

[17] 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," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 66–75.

[18] J.-R. Chang and Y.-S. Chen, "Pyramid stereo matching network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5410–5418.

[19] F. Zhang, V. Prisacariu, R. Yang, and P. H. S. Torr, "GA-net: Guided aggregation net for End-To-End stereo matching," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 185–194.

[20] X. Guo, K. Yang, W. Yang, X. Wang, and H. Li, "Group-wise correlation stereo network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3273–3282.

[21] S. Khamis, S. Fanello, C. Rhemann, A. Kowdle, J. Valentin, and S. Izadi, "StereoNet: Guided hierarchical refinement for real-time edge-aware depth prediction," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 573–590.

[22] Y. Guo, A. Yao, and Y. Chen, "Dynamic network surgery for efficient DNNs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 1379–1387.

[23] Y. He, X. Zhang, and J. Sun, "Channel pruning for accelerating very deep neural networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1389–1397.

[24] M. Courbariaux, Y. Bengio, and J.-P. David, "Binaryconnect: Training deep neural networks with binary weights during propagations," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 3123–3131.

[25] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "XNOR-Net: Imagenet classification using binary convolutional neural networks," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 525–542.

[26] Z. Liu, B. Wu, W. Luo, X. Yang, W. Liu, and K.-T. Cheng, "Bi-real net: Enhancing the performance of 1-bit cnns with improved representational capability and advanced training algorithm," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 722–737.

[27] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.

[28] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "ShuffleNet v2: Practical guidelines for efficient CNN architecture design," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 116–131.

[29] H. Pham, M. Guan, B. Zoph, Q. Le, and J. Dean, "Efficient neural architecture search via parameter sharing," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 4095–4104.

[30] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *Stat*, vol. 1050, p. 9, 2015.

[31] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[32] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[33] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2006, pp. 404–417.

[34] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2564–2571.

[35] S. T. Barnard and M. A. Fischler, "Computational stereo," *ACM Comput. Surv.*, vol. 14, no. 4, pp. 553–572, Dec. 1982.

[36] F. Zhao, Q. Huang, and W. Gao, "Image matching by normalized cross-correlation," in *Proc. IEEE Int. Conf. Acoust. Speed Signal Process.*, vol. 2, May 2006, pp. II–II.

[37] C. Lei, J. Selzer, and Y.-H. Yang, "Region-tree based stereo using dynamic programming optimization," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2006, pp. 2378–2385.

[38] S. Roy and I. J. Cox, "A maximum-flow formulation of the N-camera stereo correspondence problem," in *Proc. 6th Int. Conf. Comput. Vis.*, Jan. 1998, pp. 492–499.

[39] Y. Ruichek, "Multilevel-and neural-network-based stereo-matching method for real-time obstacle detection using linear cameras," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 1, pp. 54–62, Mar. 2005.

[40] W. Luo, A. G. Schwing, and R. Urtasun, "Efficient deep learning for stereo matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5695–5703.

[41] 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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4040–4048.

[42] J. Pang, W. Sun, J. S. Ren, C. Yang, and Q. Yan, "Cascade residual learning: A two-stage convolutional neural network for stereo matching," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 887–895.

[43] G. Yang, H. Zhao, J. Shi, Z. Deng, and J. Jia, "Segstereo: Exploiting semantic information for disparity estimation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 636–651.

[44] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: http://arxiv.org/abs/1704.04861

[45] A. Howard, M. Sandler, B. Chen, W. Wang, L.-C. Chen, M. Tan, G. Chu, V. Vasudevan, Y. Zhu, R. Pang, H. Adam, and Q. Le, "Searching for MobileNetV3," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1314–1324.

[46] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6848–6856.

[47] X. Du, M. El-Khamy, and J. Lee, "AMNet: Deep atrous multiscale stereo disparity estimation networks," 2019, *arXiv:1904.09099*. [Online]. Available: http://arxiv.org/abs/1904.09099

[48] S. Tulyakov, A. Ivanov, and F. Fleuret, "Practical deep stereo (PDS): Toward applications-friendly deep stereo matching," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 5871–5881.

[49] S. Duggal, S. Wang, W.-C. Ma, R. Hu, and R. Urtasun, "DeepPruner: Learning efficient stereo matching via differentiable PatchMatch," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4384–4393.

[50] A. Tonioni, F. Tosi, M. Poggi, S. Mattoccia, and L. D. Stefano, "Real-time self-adaptive deep stereo," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 195–204.

[51] P. Luo, Z. Zhu, Z. Liu, X. Wang, and X. Tang, "Face model compression by distilling knowledge from neurons," in *Proc. AAAI*, 2016, pp. 3560–3566.

[52] G. Chen, W. Choi, X. Yu, T. Han, and M. Chandraker, "Learning efficient object detection models with knowledge distillation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 742–751.

[53] L. Lu, M. Guo, and S. Renals, "Knowledge distillation for small-footprint highway networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2017, pp. 4820–4824.

[54] M. R. U. Saputra, P. Gusmao, Y. Almalioglu, A. Markham, and N. Trigoni, "Distilling knowledge from a deep pose regressor network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 263–272.

[55] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[56] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556

[57] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[58] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4700–4708.

[59] Y. Yang, Z. Zhong, T. Shen, and Z. Lin, "Convolutional neural networks with alternately updated clique," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2413–2422.

[60] S. Liu, S. De Mello, J. Gu, G. Zhong, M.-H. Yang, and J. Kautz, "Learning affinity via spatial propagation networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1520–1530.

[61] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.

[62] M. Menze and A. Geiger, "Object scene flow for autonomous vehicles," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3061–3070.

[63] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980

[64] Z. Liang, Y. Feng, Y. Guo, H. Liu, W. Chen, L. Qiao, L. Zhou, and J. Zhang, "Learning for disparity estimation through feature constancy," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2811–2820.

[65] R. Atienza, "Fast disparity estimation using dense networks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 3207–3212.

[66] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241.

QINQUAN GAO received the B.S. degree in automation and the M.S. degree in systems engineering from Xiamen University, China, in 2008 and 2010, respectively, and the Ph.D. degree from Imperial College London in 2014. He is currently an Associate Professor with Fuzhou University, working on model compressing, machine learning, biomedical image processing, and computer vision.

YUANBO ZHOU (Graduate Student Member, IEEE) received the B.S. degree in Internet of Things from Fuzhou University in 2018, where he is currently pursuing the master's degree. His main research interests include 3D vision, image super-resolution, and model compressing.

GEN LI received the B.S. degree in computer science and technology from Southwest University for Nationalities, Chengdu, China, in 2006, and the Ph.D. degree in electrical and electronics engineering from Yonsei University, Seoul, South Korea, in 2017. He has been a Chief Research Scientist with Imperial Vision Technology, Fuzhou, China, since 2017. His current research interests include deep learning, computer vision, image analysis, and pattern recognition.

TONG TONG received the Ph.D. degree from Imperial College London in 2015. He was a Research Fellow of the MGH/Harvard Medical School in 2016. He is currently a Full Professor with the College of Physics and Information Engineering, Fuzhou University. His research interests include machine learning, medical image analysis, and computer aided diagnosis.