

RESEARCH ARTICLE

HCVNet: Binocular Stereo Matching via Hybrid Cost Volume Computation Module With Attention

CHENGLIN DAI^{1,2}, QINGLING CHANG^{1,2}, TIAN QIU¹, XINGLIN LIU^{1,2}, AND YAN CUI^{1,2,3}¹Faculty of Intelligent Manufacturing, Wuyi University, Jiangmen 529000, China²China-Germany Artificial Intelligence Institute (Jiangmen), Wuyi University, Jiangmen 529000, China³Zhuhai 4Dage Network Technology, Zhuhai 519000, China

Corresponding author: Yan Cui (acuiyan@wyu.edu.cn)

This work was supported in part by the School Research Projects of Wuyi University under Grant 2020KZDZX1204.


ABSTRACT Binocular stereo matching, a computer vision task typically using cost volume constructed from the left and right feature maps to estimate disparity and depth, is widely applied in 3D reconstruction, autonomous driving and robotics navigation. Though recent study brings an awareness of the convolution neural networks and the attention algorithms used in this field can make great progress, it is still difficult to satisfy the demand of high-precision applications due to many reasons. Study finds that the exist methods usually incline to ignore the intermediate feature map of other scales, pay less attention to the relationship between left and right feature maps and even just tend to use one type of cost volume to train the model. In this article, we mainly focus on solving the three rproblems mentioned above. Firstly, we present the Multi-scale Feature Extraction and Fusion Module (MFEFM) to get the informational feature maps via fusing all scale feature maps. And then we design the Effective Channel Attention Module (ECAM) applied to better capture and utilize the channel-wise independencies. Finally, we adopt the Hybrid Cost Volume Computation Module (HCVCM) to construct and aggregate cost volume. With these solutions, we build an end-to-end stereo matching network named HCVNet. Comparison with other state-of-the-art models, it can achieve 0.714 EPE on SceneFlow dataset, descending PSMNet (1.09 EPE) by 37.6%.

INDEX TERMS Binocular stereo matching, feature map, channel-wise independencies, channel attention, cost volume.

I. INTRODUCTION

Binocular stereo matching, which is depth estimation in essence, usually towards getting disparity using aggregated cost volume computed from the input left and right images. According to the formula $D = B \times f / d$, the depth D would be calculated from the baseline B , the focal length f and the estimated disparity d , where baseline refers to the distance between the input left and right images. Moreover, as a classical and significant vision task, it lays a solid foundation for 3D reconstruction [1], autonomous driving [2] and robotics navigation [3].

On the one hand, since the convolution neural networks (CNNs) were introduced into the computer vision field, many

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

vision tasks based on deep learning such as image classification [4], object detection [5], object recognition [6], etc. have achieved great progress. So is the binocular stereo matching. Just taking PSMNet [7], FADNet [8] and StereoNet [9] as examples, PSMNet is a major breakthrough for integrating global context information to cost volume to address the ill-posed region problems, FADNet is implemented by 2D based correlation layers with the help of multi-scale weight training strategy to maintain faster computing speed, and StereoNet uses low-resolution cost volume to speed up running time and employs the edge sensing up sampling function to retain the details of the edge. They can indeed obtain very competitive outcomes in a certain period of time.

On the other hand, some learning-based vision tasks, such as instance segmentation [10], scene segmentation [11] and image super-resolution [12], which depend on the attention

algorithms typically used in natural language processing to focus on region of interest also perform better. Binocular stereo matching is no exception. MCANet [13] involves it for refining disparity, and NLCA-Net [14] exploits global context information with the help of it. Both of them can accomplish their targets.

However, to be exact, even though these studies can achieve compelling performance, the insufficient use problem of the other scale intermediate feature maps and the other types of cost volumes, and the less attention problem to the channel-wise independencies of the left and right feature map hinder them from meeting the requirements of high accuracy applications. Hence, in order to mitigate the impact of these above problems and gain exceedingly good effects as possible, we try to do a lot of work. Firstly, to make model learn enough useful feature map information of the original images adequately, we present the Multi-scale Feature Extraction and Fusion Module (MFEFM) to fuse all scale feature maps. Secondly, aiming to better capture and utilize the channel-wise independencies between the extracted feature map pairs, we design the Effective Channel Attention Module (ECAM). Then due to single cost volume computation method is arduous to fully employ three kinds of cost volumes, so we adopt the Hybrid Cost Volume Computation Module (HCVCM), which contains the New Cost Aggregation Module (NCAM) and the Cost Volume Construction Module (CVCM) applied to build the Hybrid Cost Volume (HCV). At last combining these solutions, we set up a competitive end-to-end stereo matching network named HCVNet. The contributions are as follows:

- We present the Multi-scale Feature Extraction and Fusion Module (MFEFM) to amply avail itself of all scale extracted feature maps from the input images.
- We design the Effective Channel Attention Module (ECAM) to sufficiently capture and resort to channel-wise independencies of the right extracted feature map according to the left extracted feature map.
- We adopt the Hybrid Cost Volume Computation Module (HCVCM) which comprises the New Cost Aggregation Module (NCAM) and the Cost Volume Construction Module (CVCM) for plentifully exploiting the Hybrid Cost Volume (HCV) and getting better results.
- We construct an end-to-end stereo matching network called HCVNet whose competitiveness can not be ignored.

II. RELATED WORK

A. LEARNING-BASED MULTI-SCALE FEATURE EXTRACTION AND FUSION APPROACH

The common learning-based multi-scale feature extraction and fusion strategies can be divided into the parallel multi branch feature extraction and fusion method and the serial skip connection structure. Without this module, the model would not be able to learn more informative messages.

1) THE PARALLEL MULTI-BRANCH FEATURE EXTRACTION AND FUSION METHOD

Typically, using the dilated convolution, altering the kernel size of convolution or utilizing the pooling operation can influence the size of the receptive field and obtain different feature maps prior to feature fusion. For example, the basic inception module in Inception V1 [15] consists of the standard 1×1 convolution, 3×3 convolution, 5×5 convolution, and 3×3 max pooling layers, which enable creating feature maps of various information. The SPP (Spatial Pyramid Pooling) structure of the SPPNet [16] can minimize the loss and deformation problem of the image information caused by a series of operations such as crop, warp, flip, and so forth as much as feasible. To gain several receptive fields and multi-scale feature maps, PSPNet [17] resorts to a technique that directly modifies the pooling procedures of different sizes, the Deeplav network [18], [19], [20] depends on the increasingly augmented ASPP (Atrus Spatial Pyramid Pooling) algorithm, and Big-Little Net [21] gets the aid of the presented parallel multi-branch network structure and fusion module.

2) THE SERIAL SKIP CONNECTION STRUCTURE

It is frequent to realize feature fusion through skipping connection. With the aid of it, U-Net [22] can ensure the final recovered feature map incorporates more low-level features of various scales, and FPN [23] can integrate the high-level features with the adjacent low-level features separately for each layer. Based on FPN, Libra R-CNN [24] produces and fuses feature maps of four scales by building a balanced feature pyramid, PAN [25] can also add the low-level features to the not adjacent higher-level layer using its feature pyramid enhancement module, ThunderNet [26] presents a simple context enhancement module to directly fuse the target feature map of three scales at a low computation cost, NAS-FPN [27] constructs a new feature fusion module with the network structure searching method, and BiFPN [28] gives birth to a feature fusion module layer with self-learning weight.

In general, these two kinds of feature extraction and fusion approaches can be appropriately chosen based on the requirement. However, the usage trend of the serial skip connection structure may rise due to it is more flexible than the parallel multi-branch feature extraction and fusion method. We support the prevailing view that sending the feature map with the local semantic information of each scale to the next task allows the network to pick up more meaningful messages and strengthen the feature representation capacity.

B. VOLUMETRIC STEREO MATCHING MODEL

Learning-based volumetric binocular stereo matching algorithms are generally divided into two categories. One is the 4D model generating cost volume with feature channel, candidate disparity, height and width, another is the 3D model constructing cost volume lacking feature channel under single resolution, where D refers to dimension.

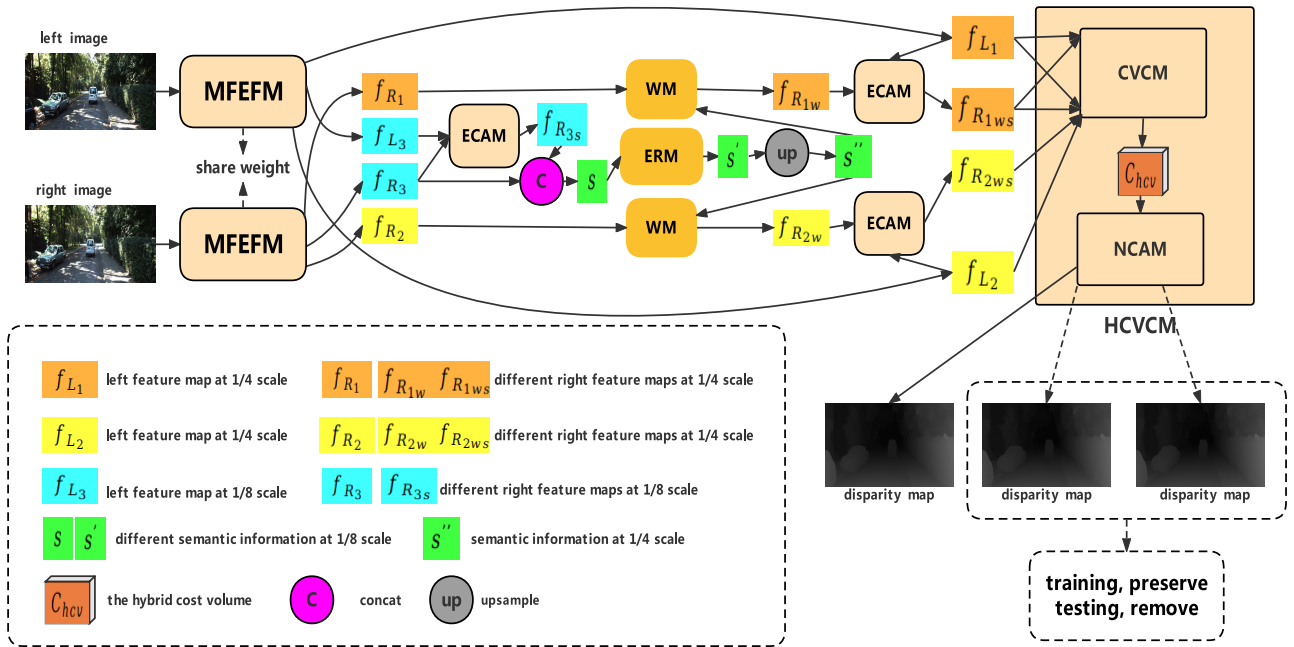


FIGURE 1. Pipeline of the proposed network HCVNet. MFEFM is the multi-scale feature extraction and fusion module. ECAM refers to the effective channel attention module. HCVCM also call the hybrid cost volume computation module, which contains the cost volume construction module (CVCM), the hybrid cost volume (HCV) C_{hcv} and the new cost aggregation module (NCAM). ERM is the edge-aware refinement module, WM is the warping module.

1) 3D MODELS

DispNetC [29] is undoubtedly a milestone owing to it is the first method to build 3D cost volume and directly regress the 3D cost volume in an end-to-end learning manner. In order to reduce the impact of ill-posed region problems, CRL [30] refines the initial disparity value with the cross-scale feature information, and AANet [31] handles them via the cross-scale cost aggregation algorithm. To maintain faster computing speed, taking FADNet as backbone, ESNet [32] simplifies the original network structure and acquires motion compensation with pixel distortion. So as to avoid the intensive calculation and memory consumption, SCV stereo [33] utilizes sparse cost volume representation to store the optimal k-value matching cost of each pixel. Thanks to those creative ideas, they are able to yield very competitive outcomes in a certain period of time. But for 3D models, the lack of enough information is what prevents them from achieving high-precision results.

2) 4D MODELS WITHOUT ATTENTION

When faced with the ill-posed region challenges, SSPCV-Net [34] extracts details from semantic segmentation sub networks, the content-aware inter-scale cost aggregation method [35] adaptively aggregates and upsamples cost volume for reliable detail recovery, and MSMD-Net [36] constructs multi-scale and multi-dimension cost volume. Moreover, on the purpose of balancing real-time performance and accuracy, Gwc-Net [37] presents a group-wise

correlation module that can not only provide similarity measurement, but also maintain better performance after reducing parameters, BGNet [38] proposes upsampling module based on the learned bilateral grid to get high quality cost volume form the low-resolution feature maps, and the method [39] realizes this goal by mean of the separable convolution. Targeting at earning a high level of accuracy, AcfNet [40] directly constraints the cost volume using true disparities peaked at unimodal distribution and the adaptive filtering cost volume. To a certain extent, these approaches are actually excellent.

3) 4D MODELS WITH ATTENTION

MA-Net [41] adaptively aggregates multi-scale context information and recalibrates hierarchical cost volumes obtained from different scales to avoid ill-posed region issues as possible. MAN [42] makes a better balance between accuracy and efficiency by way of the proposed attention module that can effectively select multi-scale information to refine the feature maps. Some experiments were done and proved that they have great development potential in some aspects.

Though these 4D models can almost achieve better performance, it is still tough to gain more accurate disparities under the influence of the issue that we concentrate on. We suppose that if the intermediate feature map at different scales are concatenated to generate the 4D hybrid cost volume and the feature map channel-wise independencies seized by the attention algorithms are used, their accuracy would be improved to some extent.

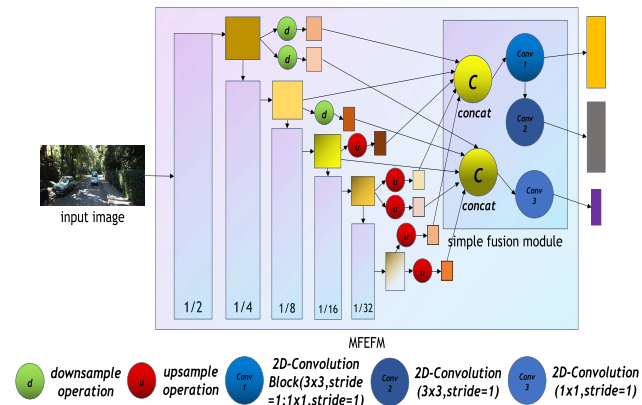


FIGURE 2. Block diagram of the MFEFM structure. The rectangles with 1/2, 1/4, 1/8, 1/16 and 1/32 scale levels jointly build the pyramid feature map extractor. The rectangles in different color and size are the different generated feature maps. 2D-Convolution Block = 2D-convolution ($3 \times 3, \text{stride}=1$) + the batch normalization + the relu activation function + 2D-convolution ($1 \times 1, \text{stride}=1$).

III. METHODOLOGY

As shown and illustrated in FIGURE 1, our model is mainly composed of the MFEFM, the ECAM and the HCVCM. In addition, it also encompasses other modules and functions. The running process of our model is briefly described as follows (as shown in FIGURE 1). After preprocessing, the image pairs are first sent to MFEFM (described in detail in Section III.A, as shown in FIGURE 2) to generate three types of different feature maps respectively. They are the left and right feature map pairs at 1/4 scale of different channels, which are denoted as f_{L1} and f_{R1} , f_{L2} and f_{R2} respectively, but the channels of the left and right feature maps are the same. Another outputs are the left and right feature maps denoted as f_{L3} and f_{R3} at 1/8 scale of the same channels. Then f_{L3} and f_{R3} are thrown into ECAM (described in detail in Section III.B, as shown in FIGURE 3) to produce a new right feature map denoted as f_{R3s} . Thirdly, f_{R3s} and f_{L3} are concatenated to create the semantic information s required by the warping module (follow [8]). Fourthly, s is transformed into s'' through the operation of the refinement module (follow [9]) and the upsample module. Fifthly, s'' is sent into the warping module to warp f_{R1} and f_{R2} , then the right feature maps after the warping operation are denoted as f_{R1w} and f_{R2w} . Sixthly, f_{L1} and f_{R1w} , f_{L2} and f_{R2w} are respectively transmitted to ECAM to originate two new right feature maps at 1/4 scale of different channels, which are denoted as f_{R1ws} and f_{R2ws} . Finally, f_{L1} and f_{R1ws} , f_{L2} and f_{R2ws} are respectively fed into the CVCM of the HCVCM (described in detail in Section III.C, as shown in FIGURE 5) to give rise to the HCV denoted as C_{hcv} and then C_{hcv} is forwarded to NCAM to yield the final disparity results.

A. MULTI-SCALE FEATURE EXTRACTION AND FUSION MODULE

Extracting feature map is the first step acting as an important role. Without sufficient information provided by the feature

extraction module, it is difficult to learn something useful for training model. If the feature information is enough but employing them insufficiently, the problem is still sitting there. So we propose the MFEFM for the binocular stereo matching with the purpose of making full use of all intermediate feature maps at different scales.

FIGURE 2 illustrates the MFEFM structure. The MFEFM consists of a special pyramid feature map extractor and a simple fusion module. We choose part of MobileNet V2 [43] as our backbone pyramid feature map extractor owing to its lightweight property. Then we build a simple fusion module using a U-Net style upsampling and downsampling operation with skip connection at each scale level. When feeding the input image which are after preprocessing into the MFEFM, subsequently, three feature maps are gotten. Different from the operation of the original feature extraction module, which generates the different scale feature maps through the general convolution and residual block, and then selects the required feature maps to fuse and produce the final feature maps, MFEFM firstly gives rise to the intermediate feature maps at 1/2, 1/4, 1/8, 1/16 and 1/32 scale respectively by the special pyramid feature map extractor, and then fuses all scale feature maps and generates the final three feature maps at 1/4, 1/4 and 1/8 scale respectively through the simple fusion module. In term of the running process brief, the final three feature maps can be f_{L1} , f_{L2} , f_{L3} or f_{R1} , f_{R2} , f_{R3} respectively.

As displayed in FIGURE 2, different from the operation of the original feature extraction module, which generates the intermediate feature maps through the general convolution and residual block, and then selects the required feature maps to fuse and generate the final feature maps, MFEFM first generates the required feature maps using part of the trained MobileNetV2 network, and then fuses and generates the final feature maps through the fuse module. During extracting feature maps, MFEFM can not generate redundant intermediate feature maps, and make full use of the generated intermediate feature maps.

In general, MFEFM can make full use of the generated intermediate feature maps at different scales and enhance the feature map representation ability of the network. Assembling it into a stereo matching model is a nice choice.

B. EFFECTIVE CHANNEL ATTENTION MODULE

In short, the attention module is the method that can redistribute the weights of the feature map content information. In other words, it can emphasize useful information and suppress useless information according to required operation. The fact that paying more attention to the vital aspect can help people understand information easily, and so is the learning-based model, triggers us to design a simple but effective channel attention module named ECAM for the binocular stereo matching to amply capture and take advantage of the channel-wise independencies of the target feature map according to the source feature map.

In essence, each channel of the feature map can be regarded as a feature detector [44], the channel attention module mainly

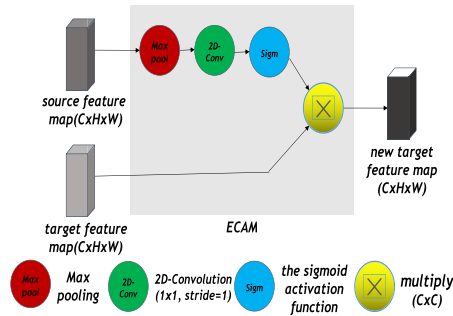


FIGURE 3. Diagram of the ECAM structure. ‘C’, ‘H’, ‘W’ refer to the channel, the height, the width of the feature map respectively.

focuses on what channel content or channel information is meaningful of the given input image. To avoid the heavy computation burden brought by the complex attention module commonly adopted in learning-based tasks, as displayed in FIGURE 3, we build the ECAM with only a max pooling function, a 2D convolution, a sigmoid activated function and a multiply operation even though these complex attention module can make model reach quite good effects. Besides, the average pooling function is usually used in common attention module, but the thought [45] that the max pooling function gathers another important clue about distinctive object features to infer finer channel-wise attention arouses us to select the max pooling function to improve the network representation capability. The example in FIGURE 4 shows the effect of the ECAM. There is no doubt that ECAM works. It also means that in the light of the source feature map, some edge or shape cues of the target feature map can be focused on to emphasize.

The ECAM running process can be defined as:

$$f_{i'} = ECAM(f_s, f_t) \tag{1}$$

where f_s is the source feature map, f_t is the target feature map, and $f_{i'}$ is the new target feature map.

In the light of the previous short running process description, we can know f_{L1} and f_{R1w} , f_{L2} and f_{R2w} , f_{L3} and f_{R3} are fed into ECAM to yield f_{R1ws} , f_{R2ws} and f_{R3s} respectively. In addition, when training HCVNet, we let f_{L1} , f_{L2} and f_{L3} act as the source feature map and multiplied factor operated with the target feature map f_{R1w} , f_{R2w} and f_{R3} respectively.

C. HYBRID COST VOLUME COMPUTATION MODULE

Generally speaking, the cost volume construction and aggregation module are tightly-coupled jointly determining the accuracy and efficiency of a stereo matching network. Thus we adopt the HCVCV which embraces the CVCM applied to erect the HCV, and the NCAM utilized to offer a suitable platform to adequately exploit the HCV.

There are three categories of cost volumes typically used in 4D stereo matching model, which are the subtract cost volume, the group-wise correlation cost volume and the concat cost volume respectively. The model armed with the subtract cost volume can obtain outcomes faster with useful

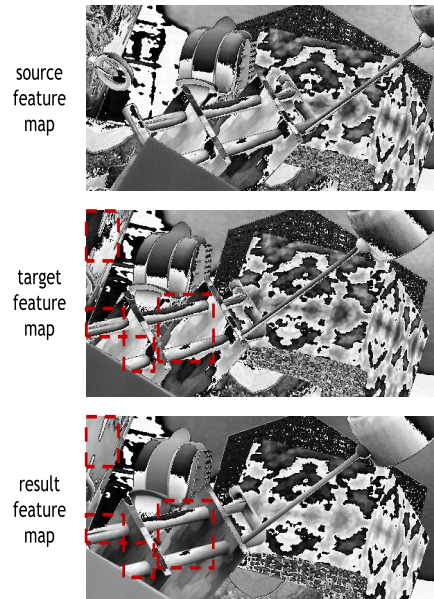


FIGURE 4. Effect diagram of the ECAM. The red box circled area indicates the part with outstanding changes before and after the module operation.

difference information between the input image pairs. Besides, the group-wise correlation cost volume is able to store the average information of the image pairs channel in groups and let the results stabilize within a certain range. When mentioning the concat cost volume, why it enjoys the great popularity is that it offers all image feature map information as possible. Inspired by them, we decide to combine them in concatenating way to sufficiently employ them and absorb their merits. So as displayed in FIGURE 5, the CVCM firstly generates the subtract cost volume, the concat cost volume and group-wise correlation cost volume respectively according to the input feature maps, and then concatenates three varieties of the cost volumes to the HCV denoted as C_{hcv} .

Aggregating cost volume is also the most important part in stereo matching network. Just because using previous cost volume aggregation module of backbone model is not suitable to the HCV, so we modify the original cost aggregation module to the NCAM to give full play to the HCV. As displayed in FIGURE 5, in our HCVNet model, the cost aggregation architecture not only follows Gwc-Net with three 3D hourglass modules, but also adds a 3D-Convolution Block to reduce the computational cost. When C_{hcv} is fed to the NCAM, the 3D-Convolution Block would diminish the number of channels from 136 to 64, then the first hourglass module would lessen the number of channels from 64 to 32, while other two hourglass modules do not need to execute the operation. During this process, C_{hcv} is iterated for many times for fully filtering and utilizing the information. When training our model, we not only send the final result C_{hcv3} to the disparity regression module to generate the final disparity result, but also feed the intermediate results C_{hcv1} and C_{hcv2} into the disparity regression module to produce disparity outputs to avoid wasting useful information.

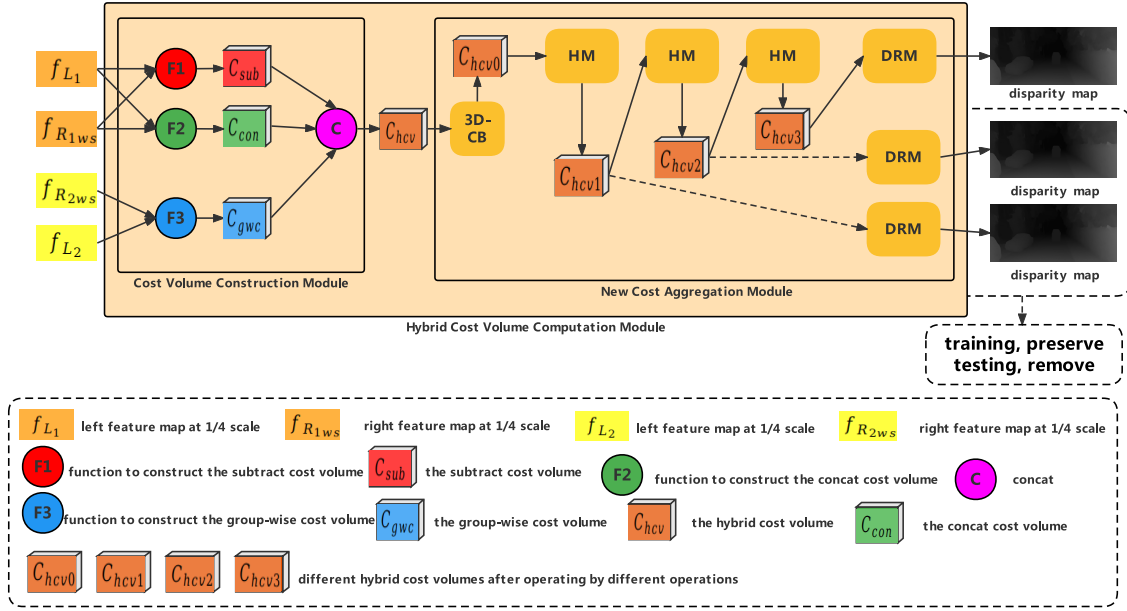


FIGURE 5. Overall architecture of the hybrid cost volume computation module (HCVCM), which includes the cost volume construction module (CVCVM), the hybrid cost volume (HCV) C_{hcv} and the new cost aggregation module (NCAM). HM is the Hourglass Module, DRM is the Disparity Regression Module. 3D-CB refers to the 3D-Convolution Block. 3D-Convolution Block = 3D-Convolution ($3 \times 3 \times 3$, stride=1) + batch normalization + the relu activation function.

In line with the running process brief, the whole process of aggregating cost volume can be roughly summarized as:

$$C_{hcv3} = NCAM(C_{hcv}) \quad (2)$$

where C_{hcv3} represents the final result and C_{hcv} refers to the initial input HCV. And the process of building C_{hcv} can be described as:

$$C_{hcv} = C_{sub} \parallel C_{con} \parallel C_{gwc} \quad (3)$$

where C_{sub} means the subtract cost volume, C_{gwc} refers to the group-wise correlation cost volume, C_{con} represents the concat cost volume and \parallel is the concatenation operation.

In addition, according to the running process brief, these cost volumes at pixel location (x, y) using the feature maps at scale level $s = 1/4$ are separately computed as:

$$C_{sub}(d, x, y, f) = f_{L1}(x, y) - f_{R1ws}(x - d, y) \quad (4)$$

$$C_{con}(d, x, y, f) = f_{L1}(x, y) \parallel f_{R1ws}(x - d, y) \quad (5)$$

$$C_{gwc}(d, x, y, g) = \frac{1}{N_c/N_g} \langle f_{L2}(x, y), f_{R2ws}(x - d, y) \rangle \quad (6)$$

where d is an integer within the maximum candidate disparity search range, i.e., $d \in (0, D_{max}/4]$ and f is the ordinal number meaning the f_{th} channel. f_{L1} and f_{L2} represent the extracted left feature maps, f_{R1ws} and f_{R2ws} are the extracted right feature maps after operating by the warping module and the ECAM. N_c refers to the channel number of the extracted feature map, and N_g is the number of groups, then each feature group therefore has N_c/N_g channels. g is the ordinal number meaning the g_{th} feature group which contains the

$gN_c/N_g, gN_c/N_g + 1, \dots, gN_c/N_g + (N_c/N_g - 1)_{th}$ channels of the input feature maps. $\langle *, * \rangle$ is inner product and \parallel denotes the concatenation operation.

D. DISPARITY REGRESS FUNCTION

The disparity regression, which is used to predict the continuous disparity maps proposed in [46], is more robust than classification-based regress functions. The equation is defined as:

$$\hat{d} = \sum_{d=0}^{D_{max}-1} d \times \sigma(C_{hcv3}) \quad (7)$$

where D_{max} is set to 192, which refers to the maximum range. And for each pixel, the softmax operation $\sigma(*)$ can calculate the probability of each candidate disparity d from the aggregated hybrid cost volume C_{hcv3} . Then the final estimated disparity result \hat{d} is the sum of each candidate disparity weighted by its probability.

E. LOSS FUNCTION

Compared to the L2 loss, the smooth L1 loss [47] is widely used for its robustness and low sensitivity to outliers. So the loss function of our model is defined as:

$$L(d, \hat{d}) = \frac{1}{N} \sum_{i=1}^N smooth_{L1}(d_i - \hat{d}_i) \quad (8)$$

and $smooth_{L1}(*)$ is defined as:

$$smooth_{L1}(x) = \begin{cases} 0.5 x^2, & \text{if } |x| < 1 \\ |x| - 0.5, & \text{otherwise} \end{cases} \quad (9)$$

TABLE 1. Comparison with other state-of-the-arts models. Bold: Best. Underscore: Second best. ‘-’: Not done. ‘KIT12’: KITTI 2012 dataset. ‘KIT15’: KITTI 2015 dataset.

Method	SceneFlow EPE ↓	KIT12 3px(%)↓		KIT15 D1-all(%)↓		avg run time(s)↓
		Noc	All	Noc	All	
PSMNet	1.09	1.49	1.89	2.14	2.32	0.41
StereoNet	1.101	—	—	—	4.83	0.015
Gwc-Net	<u>0.765</u>	1.32	<u>1.70</u>	<u>1.92</u>	<u>2.11</u>	0.32
FADNet	0.83	—	—	2.59	2.82	0.048
AcfNet	0.87	1.17	1.54	1.72	1.89	—
BGNet+	1.17	1.62	2.03	—	2.19	<u>0.032</u>
HCVNet	0.714	<u>1.31</u>	1.72	2.00	2.19	0.26

where N is the total number of the labeled image pairs, d is the ground-truth disparity, and \hat{d} is the predicted disparity.

IV. EXPERIMENTS

A. DATASETS AND EVALUATION METRICS

We use three popular and public datasets [2], [29], [48] for training and finetuning our model. These datasets are introduced as follows.

SceneFlow: The SceneFlow dataset is a large synthetic stereo dataset which contains 35,454 training and 4,370 testing image pairs in 960×540 pixels resolution. It is large enough for directly training deep learning models with accurate and high-quality dense ground-truth disparity maps.

KITTI Stereo 2015: The KITTI Stereo 2015 dataset is real-world dataset with street views captured from a driving car. It is composed of 200 training and 200 testing image pairs of 1242×375 pixels resolution with sparse ground truth disparities obtained from LiDAR and fitted 3D CAD models.

KITTI Stereo 2012: The KITTI Stereo 2012 dataset is a real-world dataset with dynamic street and road views. It consists of 194 training image pairs with sparse ground truth disparities and 195 testing image pairs without ground truth. Those image pairs are in 1226×370 pixels resolution.

And the metrics that we use to evaluate methods are introduced as follows.

EPE: It is also called end-point-error, which refers to the mean average disparity error in all pixels.

DI: It means that the percentage of stereo disparity outliers in first frame. In other words, it is the error pixels that exceeds 3px and exceeds 5% of the truth value.

DI-all: It is the percentage of stereo disparity outliers averaged over all ground truth pixels.

DI-bg: It is the percentage of stereo disparity outliers averaged only over background regions.

DI-fg: It is the percentage of stereo disparity outliers averaged only over foreground regions.

xpx: It is also called xpx-error, which can be regarded as the percentage of the error pixels that exceeds xpx. The x is usually set to 2, 3, 4 or 5.

Out-Noc: It is the percentage of the error pixels in non-occluded areas.

Out-All: It is the percentage of the error pixels in all areas.

Avg-Noc: It also refers to the mean error, and it is the average disparity or end-point error in non-occluded areas.

Avg-All: It also refers to the mean error, and it is average disparity or end-point error in all areas.

TABLE 2. Comparison with other state-of-the-arts models in non-occluded areas on KITTI 2012 dataset. Bold: Best. Underscore: Second best. ‘-’: Not done.

Method	2px(%)↓	3px(%)↓	5px(%)↓	mean error(px)↓
PSMNet	2.44	1.49	0.90	0.5
StereoNet	4.91	—	—	<u>0.8</u>
Gwc-Net	<u>2.16</u>	1.32	0.80	0.5
AcfNet	1.83	1.17	0.77	—
BGNet+	2.78	1.62	—	0.5
HCVNet	<u>2.16</u>	<u>1.31</u>	<u>0.79</u>	0.5

TABLE 3. Comparison with other state-of-the-arts models in all areas on KITTI 2012 dataset. Bold: Best. Underscore: Second best. ‘-’: Not done.

Method	2px(%)↓	3px(%)↓	5px(%)↓	mean error(px)↓
PSMNet	3.01	1.89	1.15	<u>0.6</u>
StereoNet	6.02	—	—	0.9
Gwc-Net	<u>2.71</u>	<u>1.70</u>	<u>1.03</u>	0.5
AcfNet	2.35	1.54	1.01	—
BGNet+	3.35	2.03	—	<u>0.6</u>
HCVNet	2.75	1.72	1.05	0.5

B. IMPLEMENTATION DETAILS

We choose the MobileNetV2 pre-trained on ImageNet [49] as our feature map extractor backbone due to its less parameters and stronger learning ability which can make model converge faster during training. We implement our model by avail of PyTorch and choose Adam optimizer [50] as our optimizer. We randomly crop images to size $W = 512, H = 256$ for training.

On the SceneFlow dataset, we train our network for first 6 epochs with a learning rate 1×10^{-3} and then set the learning rate to 1×10^{-4} for last 4 epochs. We also set the batch size to 12 during training.

For our experiments on the KITTI dataset, we finetune the model which is pre-trained on the SceneFlow dataset for first 200 epochs with an initial learning rate of 1×10^{-3} and then decrease learning rate to 1×10^{-4} for last 100 epochs. And we set batch size to 8.

C. MODEL PERFORMANCE

We show the comparisons of our method and other existing state-of-the-art methods from TABLE 1 to TABLE 4, which can inform us straightforwardly of the fact that HCVNet’s performance is very competitive. And we show the visual results in FIGURE 6, FIGURE 7 and FIGURE 8, which can directly illustrate HCVNet perform much better than PSMNet according to this visual results.

It can be seen more intuitively from TABLE 1 that HCVNet gets great scores on SceneFlow dataset and KITTI 2012 dataset indeed. In addition, all the evaluation indicators

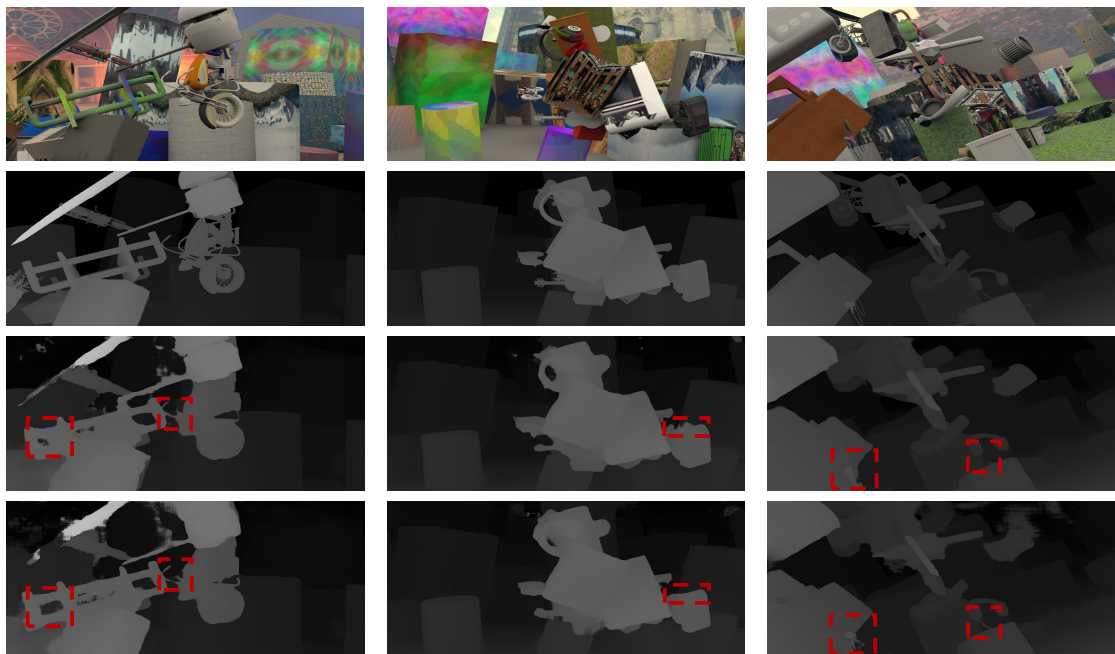


FIGURE 6. Qualitative test results on the SceneFlow dataset. From top to bottom, they are left image, ground-truth disparity map, disparity map generated by PSMNet and disparity map generated by HCVNet. The red box circled area indicates the part with outstanding changes, meaning that better edges or shapes are produced.

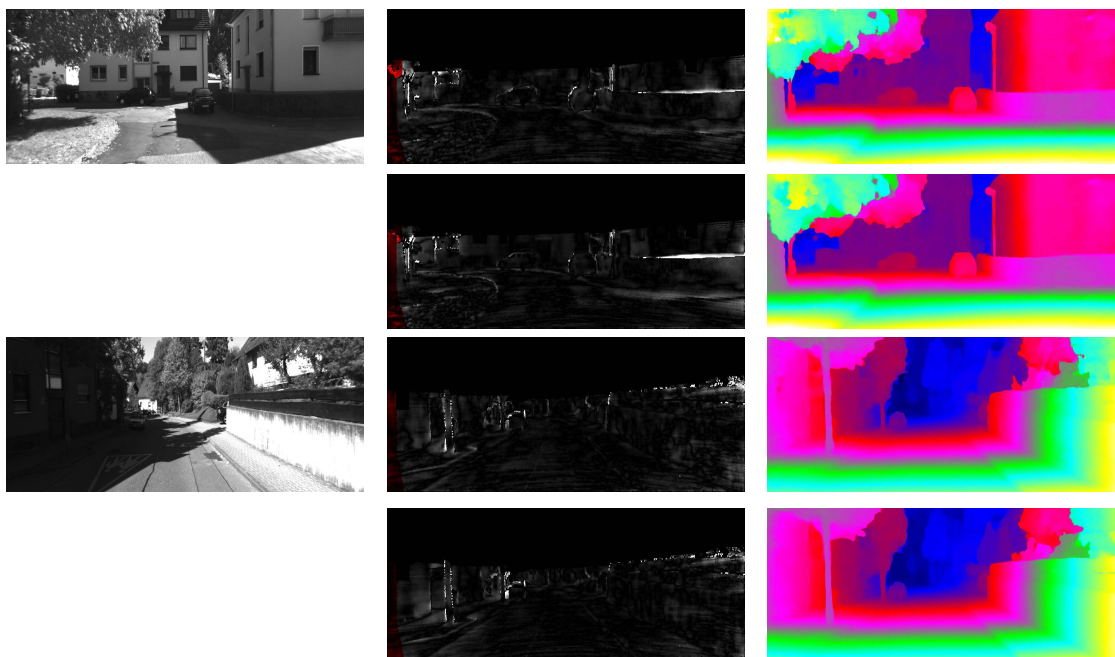


FIGURE 7. Qualitative test results on the KITTI Stereo 2015 dataset. From left to right, they are left image, disparity map, error map respectively. In the middle and right column, from the first line to the second line, and from the third line to the last line, they are the results generated by PSMNet and the outputs generated by HCVNet respectively. The blue color tones in error map means correct estimates, while the red color tones refers to wrong estimates. Dark regions in the error images denote the occluded pixels which fall outside the image boundaries.

diminish significantly. On SceneFlow dataset, HCVNet is more outstanding due to its EPE is 0.714, and Gwc-Net whose EPE is 0.765 just ranks second. They lessen PSMNet’s EPE by 0.376 and 0.325. Besides, on KITTI 2012 dataset, our

model achieves superior performance owing to 1.31% 3px ranks second in non-occluded areas. Just taking 3px on KITTI 2012 dataset as example, HCVNet lets PSMNet decline by 0.18% in non-occluded areas and 0.17% in all areas.

TABLE 4. Comparison with other state-of-the-arts models on KITTI 2015 dataset. Bold: Best. Underscore: Second best. ‘-’: Not done.

Method	All(%)			Noc(%)		
	D1-bg↓	D1-fg↓	D1-all↓	D1-bg↓	D1-fg↓	D1-all↓
PSMNet	1.86	4.62	2.32	1.71	4.31	2.14
StereoNet	4.30	7.45	4.83	—	—	—
Gwc-Net	1.74	<u>3.93</u>	<u>2.11</u>	1.61	<u>3.49</u>	<u>1.92</u>
FADNet	2.68	3.50	2.82	2.49	3.07	2.59
AcfNet	—	—	1.89	—	—	1.72
BGNet+	<u>1.81</u>	4.09	2.19	—	—	—
HCVNet	<u>1.81</u>	4.07	2.19	<u>1.68</u>	3.65	2.00

From TABLE 1 to TABLE 4 especially TABLE 4, we can conclude that HCVNet has a lot of room for improvement if keeping carry on research in summary. Though our model HCVNet is not better than other networks in all in other metrics validation, nevertheless, HCVNet has great advance over PSMNet.

All in all, though compared with other models, the avg run time of our network still has a room to upgrade, the experiments done by us yet can prove that our study is meaningful and useful because our model’s results own certain competitiveness indeed.

D. ABLATION STUDY

In order to analyse the performance of our proposed modules correctly, we train and test PSMNet again in the same environment, which is denoted as PSMNet*. And we purpose to further verify the performance of HCV and NCAM in HCVCM, so we separately replace the original cost volume and the original cost aggregation module of PSMNet. We also separately discard HCV and NCAM of HCVNet for ablation study. The ablation experiment results are shown from TABLE 5 to TABLE 8.

On the one hand, in all ablation study tables, we can see that when PSMNet* is armed correspondingly with substituting modules, although the evaluation metrics on each dataset is not better than that without replacing the original module, their overall output nearly tend to be better. On the other hand, we can also know that the performance of HCVNet would be worse and worse while uninstalling corresponding components we adopted.

In TABLE 5, it can be seen that when PSMNet* is equipped with HCVCM, the EPE drops from 1.111 to 0.904, while training and testing HCVNet without HCVCM, the EPE increases from 0.714 to 1.037. And from TABLE 6 and TABLE 7, we can know all the values are almost best while replacing with the HCVCM and testing on KITTI 2012 dataset, but when the model running without HCVCM, the results would be worse and worse. When mentioning HCV or NCAM alone from TABLE 5 to TABLE 8, its effect would result in a better degree if HCV and NCAM jointly cooperating.

TABLE 5. Ablation study results comparison. ‘+’: Adopt corresponding modules. ‘-’: Use original modules. Bold: Best. Underscore: Second best. ‘KT12’: KITTI 2012 dataset. ‘KT15’: KITTI 2015 dataset.

Method	SceneFlow EPE ↓	KT12 3px(%)↓		KT15 D1-all(%)↓	
		Noc	All	Noc	All
PSMNet*	1.111	1.48	1.91	2.27	2.49
+MFEFM	1.048	1.54	1.93	2.24	2.45
+ECAM	1.071	1.50	1.90	2.32	2.52
+HCV	1.06	1.41	<u>1.79</u>	<u>2.14</u>	2.37
+NCAM	<u>0.91</u>	<u>1.40</u>	1.80	2.02	2.23
+HCVCM	0.904	1.37	1.78	2.15	<u>2.36</u>
HCVNet	0.714	1.31	1.72	2.00	2.19
-MFEFM	0.886	<u>1.37</u>	1.79	2.15	2.35
-ECAM	0.895	<u>1.37</u>	<u>1.78</u>	<u>2.08</u>	<u>2.28</u>
-HCV	0.909	1.53	1.95	2.23	2.43
-NCAM	1.014	1.55	1.96	2.26	2.51
-HCVCM	1.037	1.61	2.04	2.47	2.69

TABLE 6. Ablation study results comparison in non-occluded areas on KITTI 2012 dataset. ‘+’: Adopt corresponding modules. ‘-’: Use original modules. Bold: Best. Underscore: Second best.

Method	2px(%)↓	3px(%)↓	4px(%)↓	5px(%)↓	mean error(px)↓
PSMNet*	2.50	1.48	1.09	0.88	0.5
+MFEFM	2.62	1.54	1.13	0.91	0.5
+ECAM	2.53	1.50	1.11	0.88	0.5
+HCV	2.38	1.41	<u>1.04</u>	<u>0.83</u>	0.5
+NCAM	<u>2.28</u>	<u>1.40</u>	<u>1.04</u>	0.84	0.5
+HCVCM	2.26	1.37	1.01	0.80	0.5
HCVNet	2.16	1.31	0.98	0.79	0.5
-MFEFM	<u>2.26</u>	<u>1.37</u>	1.03	0.83	0.5
-ECAM	2.28	<u>1.37</u>	<u>1.02</u>	<u>0.82</u>	0.5
-HCV	2.54	1.53	1.13	0.91	0.5
-NCAM	2.63	1.55	1.14	0.90	0.5
-HCVCM	2.70	1.61	1.18	0.94	0.5

In addition, we can be aware of that MFEFM can boost performance from all ablation tables. When equipping with MFEFM, PSMNet*+MFEFM decreases PSMNet* to 1.048 EPE. When not equipping with MFEFM, HCVNet-MFEFM increases HCVNet’s EPE to 0.886. And taking PSMNet* and PSMNet*+MFEFM in TABLE 8 as examples, not only in all areas D1-fg drops from 5.20% to 4.60% and D1-all decreases from 2.49% to 2.45%, but also in non-occluded areas D1-fg declines from 4.75% to 4.20% and D1-all diminishes from 2.27% to 2.24%. If we take HCVNet and HCVNet-MFEFM as instances, we can know not only in non-occluded areas 3px raises from 1.31% to 1.37% and D1-all grows from 2.00% to 2.15%, but also in all areas 3px ascends from 1.72% to 1.79% and D1-all expands from 2.19% to 2.35%.

Furthermore, when equipping with ECAM on SceneFlow dataset, PSMNet*+ECAM’s EPE is 1.071 that decreases PSMNet* by 0.04 EPE, and HCVNet-ECAM’s EPE is 0.895 that increases HCVNet’s EPE by 0.181. Besides, in all areas on KITTI 2012 dataset, 3px and 5px are 1.90% and 1.13% respectively, making PSMNet* descend by 0.01%. At the same time, from TABLE 5 to TABLE 8, we also can be informed that when not arming with ECAM, in non-occluded

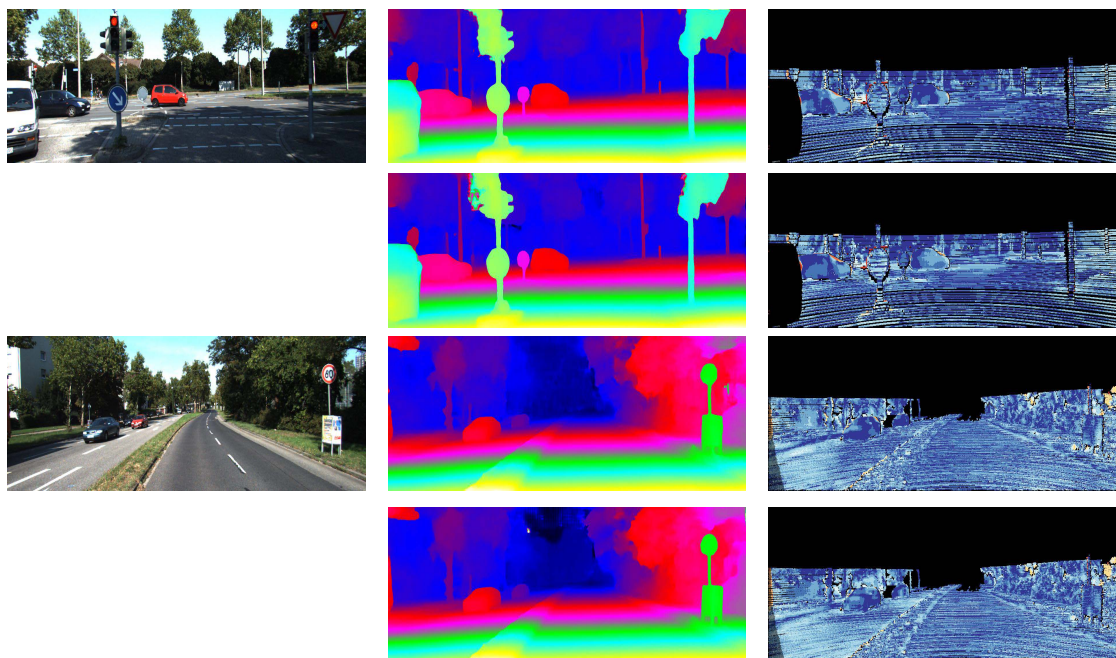


FIGURE 8. Qualitative test results on the KITTI Stereo 2012 dataset. From left to right, they are left image, error map, disparity map respectively. In the middle and right column, from the first line to the second line, and from the third line to the last line, they are the results generated by PSMNet and the outputs generated by HCVNet respectively. The error map scales linearly between 0 (black) and ≥ 5 (white) pixels error. The red color in error map denotes all occluded pixels, falling outside the image boundaries.

TABLE 7. Ablation study results comparison in all areas on KITTI 2012 dataset. ‘+’: Adopt corresponding modules. ‘-’: Use original modules. **Bold: Best. Underscore: Second best.**

Method	2px(%)↓	3px(%)↓	4px(%)↓	5px(%)↓	mean error(px)↓
PSMNet*	3.10	1.91	1.42	1.14	<u>0.6</u>
+MFEFM	3.20	1.93	1.43	1.16	<u>0.6</u>
+ECAM	3.11	1.90	1.41	1.13	<u>0.6</u>
+HCV	2.93	<u>1.79</u>	<u>1.32</u>	<u>1.07</u>	0.5
+NCAM	<u>2.87</u>	1.80	1.34	1.08	0.5
+HCVCM	2.84	1.78	1.31	1.04	0.5
HCVNet	2.75	1.72	1.29	1.05	0.5
-MFEFM	2.86	1.79	<u>1.34</u>	<u>1.08</u>	0.5
-ECAM	<u>2.86</u>	1.78	<u>1.34</u>	<u>1.08</u>	<u>0.6</u>
-HCV	3.14	1.95	1.45	1.17	<u>0.6</u>
-NCAM	3.22	1.96	1.45	1.15	<u>0.6</u>
-HCVCM	3.32	2.04	1.50	1.19	<u>0.6</u>

areas and all areas on KITTI 2012 dataset, 3px are 1.37% and 1.78% respectively, making HCVNet go up to 1.31% and 1.72%. And in non-occluded areas and all areas on KITTI 2015 dataset, D1-all are 2.08% and 2.28% respectively, ascending HCVNet by 0.08% and 0.09%. Then we can have knowledge of that ECAM are able to improve model even if the effect is not obvious.

We can be informed that HCVCVM combines and gives full play to the advantages of HCV and NCAM from these tables. Just taking PSMNet* and PSMNet*+HCVCVM in TABLE 6 as instances, in non-occluded areas 2px dwindles from 2.50% to 2.26%, 3px declines from 1.48% to 1.37%, 4px lessens from 1.09% to 1.01%, and 5px diminishes from

TABLE 8. Ablation study results comparison on KITTI 2015 dataset. ‘+’: Adopt corresponding modules. ‘-’: Use original modules. **Bold: Best. Underscore: Second best.**

Method	All(%)			Noc(%)		
	D1-bg↓	D1-fg↓	D1-all↓	D1-bg↓	D1-fg↓	D1-all↓
PSMNet*	1.95	5.20	2.49	1.78	4.75	2.27
+MFEFM	2.02	4.60	2.45	1.86	4.20	2.24
+ECAM	1.99	5.20	2.52	1.83	4.82	2.32
+HCV	1.93	<u>4.57</u>	2.37	1.77	<u>4.03</u>	<u>2.14</u>
+NCAM	1.85	4.11	2.23	1.69	3.72	2.02
+HCVCVM	<u>1.91</u>	4.64	<u>2.36</u>	<u>1.75</u>	4.20	2.15
HCVNet	1.81	4.07	2.19	1.68	3.65	2.00
-MFEFM	1.92	4.52	2.35	1.77	4.08	2.15
-ECAM	<u>1.85</u>	<u>4.40</u>	<u>2.28</u>	<u>1.69</u>	<u>4.04</u>	<u>2.08</u>
-HCV	2.03	4.45	2.43	1.87	4.08	2.23
-NCAM	2.05	4.84	2.51	1.89	4.14	2.26
-HCVCVM	2.13	5.52	2.69	1.95	5.08	2.47

0.88% to 0.80%. We also can see that using HCVCVM is better than utilizing HCV or NCAM alone. It is also clear that they are closely linked and indispensable. The NCAM provide the most suitable platform for aggregating the HCV, while the HCV is rich in entire kinds of the feature map information. With only one module, the expression and representation ability of the model is limited. So we can see that the performances of PSMNet*+HCV and PSMNet*+NCAM are poor when comparing with PSMNet*+HCVCVM. And we consider that the reason why PSMNet*+HCVCVM performs much better than PSMNet* but is worse than PSMNet*+NCAM is

that NCAM happens to be one of the most suitable platforms for the original cost volume built by PSMNet* to aggregate itself. Meanwhile, HCVNet-HCVCM performs much worse than HCVNet-HCV and HCVNet-NCAM, which can be explained from the side that compared with HCV and NCAM, HCVCM has a greater impact on the network, meaning that HCVCM can significantly improve the performance of the model.

In brief, the adopted modules can augment the model performance.

V. CONCLUSION

Our goal is getting more accurate results and mitigating the impact of the inadequately use problem of the other scale intermediate feature maps and the other types of cost volumes, and the less attention problem to the channel-wise independencies of the left and right feature map. Thus, we propose the MFEFM and the ECAM for the binocular stereo matching and adopt the HCVCM to solve these problem as possible. We also construct a model called HCVNet for binocular stereo matching, and do some experiments to validate its superiority over other state-of-the-art methods in this paper. Although compared with them, our model can not do the best in all aspects, such as avg run time, its competitive performance (0.714 EPE on SceneFlow dataset, 1.31% 3px in non-occluded areas on KITTI 2012 dataset and 2.00% D1-all in non-occluded areas on KITTI 2015 dataset) in a way should not be ignored. After all, the avg run time of HCVNet (0.26s) under the running condition (four NVIDIA Tesla V100 GPUs) is 0.15s less than the backbone model PSMNet (0.41s). It can draw a conclusion that the adopted components (MFEFM, ECAM and HCVCM) are effective and useful in binocular stereo matching. It is very hopeful that the method can be beneficial to various vision tasks. Besides, our next ambition is planing to promote efficiency and maintain model performance meanwhile as possible.

REFERENCES

- [1] Z. Gao, E. Li, Z. Wang, G. Yang, J. Lu, B. Ouyang, D. Xu, and Z. Liang, "Object reconstruction based on attentive recurrent network from single and multiple images," *Neural Process. Lett.*, vol. 53, no. 1, pp. 653–670, Feb. 2021.
- [2] 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.
- [3] C. Luo, L. Yu, E. Yang, H. Zhou, and P. Ren, "A benchmark image dataset for industrial tools," *Pattern Recognit. Lett.*, vol. 125, pp. 341–348, Jul. 2019.
- [4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 25, Dec. 2012, pp. 1097–1105.
- [5] H. Zheng, J. Chen, L. Chen, Y. Li, and Z. Yan, "Feature enhancement for multi-scale object detection," *Neural Process. Lett.*, vol. 51, no. 2, pp. 1907–1919, Apr. 2020.
- [6] C. Yu, B. Xiao, C. Gao, L. Yuan, L. Zhang, N. Sang, and J. Wang, "LiteHRNet: A lightweight high-resolution network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 10440–10450.
- [7] 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.
- [8] Q. Wang, S. Shi, S. Zheng, K. Zhao, and X. Chu, "FADNet: A fast and accurate network for disparity estimation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 101–107.
- [9] S. Khamsi, 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.
- [10] Y. Lee and J. Park, "CenterMask: Real-time anchor-free instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 13906–13915.
- [11] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3146–3154.
- [12] L. Wang, Y. Wang, Z. Liang, Z. Lin, J. Yang, W. An, and Y. Guo, "Learning parallax attention for stereo image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12250–12259.
- [13] H. Sang, Q. Wang, and Y. Zhao, "Multi-scale context attention network for stereo matching," *IEEE Access*, vol. 7, pp. 15152–15161, 2019.
- [14] Z. Rao, M. He, Y. Dai, Z. Zhu, B. Li, and R. He, "NLCA-Net: A non-local context attention network for stereo matching," *APSIPA Trans. Signal Inf. Process.*, vol. 9, no. 1, pp. 1–13, 2020.
- [15] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, and D. Erhan, "Vincent Vanhoucke, Andrew Rabinovich," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.
- [16] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2014.
- [17] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2881–2890.
- [18] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2017.
- [19] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*.
- [20] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 801–818.
- [21] C.-F. Chen, Q. Fan, N. Mallinar, T. Sercu, and R. Feris, "Big-little Net: An efficient multi-scale feature representation for visual and speech recognition," 2018, *arXiv:1807.03848*.
- [22] 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*, 2015, pp. 234–241.
- [23] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 2117–2125.
- [24] J. Pang, K. Chen, J. Shi, H. Feng, W. Ouyang, and D. Lin, "Libra R-CNN: Towards balanced learning for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 821–830.
- [25] W. Wang, E. Xie, X. Song, Y. Zang, W. Wang, T. Lu, G. Yu, and C. Shen, "Efficient and accurate arbitrary-shaped text detection with pixel aggregation network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8440–8449.
- [26] Z. Qin, Z. Li, Z. Zhang, Y. Bao, G. Yu, Y. Peng, and J. Sun, "ThunderNet: Towards real-time generic object detection on mobile devices," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6718–6727.
- [27] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "NAS-FPN: Learning scalable feature pyramid architecture for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7036–7045.
- [28] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10781–10790.
- [29] 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.
- [30] 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.

- [31] H. Xu and J. Zhang, "AANet: Adaptive aggregation network for efficient stereo matching," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1959–1968.
- [32] Z. Huang, T. B. Norris, and P. Wang, "ES-Net: An efficient stereo matching network," 2021, *arXiv:2103.03922*.
- [33] H. Wang, R. Fan, and M. Liu, "SCV-Stereo: Learning stereo matching from a sparse cost volume," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2021, pp. 3203–3207.
- [34] Z. Wu, X. Wu, X. Zhang, S. Wang, and L. Ju, "Semantic stereo matching with pyramid cost volumes," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7484–7493.
- [35] C. Yao, Y. Jia, H. Di, Y. Wu, and L. Yu, "Content-aware inter-scale cost aggregation for stereo matching," 2020, *arXiv:2006.03209*.
- [36] Z. Shen, Y. Dai, and Z. Rao, "MSMD-Net: Deep stereo matching with multi-scale and multi-dimension cost volume," 2020, *arXiv:2006.12797*.
- [37] 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.
- [38] B. Xu, Y. Xu, X. Yang, W. Jia, and Y. Guo, "Bilateral grid learning for stereo matching networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 12497–12506.
- [39] R. Rahim, F. Shamsafar, and A. Zell, "Separable convolutions for optimizing 3D stereo networks," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2021, pp. 3208–3212.
- [40] Y. Zhang, Y. Chen, X. Bai, S. Yu, K. Yu, Z. Li, and K. Yang, "Adaptive unimodal cost volume filtering for deep stereo matching," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 7, pp. 12926–12934.
- [41] L. Guo, H. Duan, and W. Zhou, "Multiple attention networks for stereo matching," *Multimedia Tools Appl.*, vol. 80, no. 18, pp. 28583–28601, 2021.
- [42] X. Yang, L. He, Y. Zhao, H. Sang, Z. L. Yang, and X. J. Cheng, "Multi-attention network for stereo matching," *IEEE Access*, vol. 8, pp. 113371–113382, 2020.
- [43] 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.
- [44] D. M. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 818–833.
- [45] S. Woo, J. Park, J.-Y. Lee, and A. I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.
- [46] 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.
- [47] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.
- [48] 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.
- [49] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
- [50] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.



QINGLING CHANG received the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, in 2015. She is currently a Master Supervisor and an Associate Professor with Wuyi University and the Subdecanal of the China-German Artificial Intelligence Institute. She has published more than ten papers, included in SCI/EI. In this paper, she is mainly responsible for the overall framework design. Her research interests include artificial intelligence, computer vision, and knowledge graph.



TIAN QIU received the B.Eng. degree in measurement and instrumentation and the M.Sc. and Ph. D. degrees in circuits and systems from the University of Science and Technology of China, in 2000, 2003 and 2006, respectively. He worked as an Engineer and a Senior Engineer with Samsung Electronics, South Korea, from 2006 to 2009; a Research Associate with the University of Kent, U.K., from 2009 to 2012; and a Leading Engineer and a Research Engineer with Imagination Technologies, U.K., from 2012 to 2016. He is currently a Contract Professor of image processing with the School of Intelligent Manufacture, Wuyi University. His research interests include image processing, image analysis, and computer vision.



XINGLIN LIU received the Master of Computer Technology degree from the School of Computer, Chongqing University, in December 2005, and the Ph.D. degree in applied computer technology from the School of Computer Science and Engineering, South China University of Technology, in June 2012. He is currently an Associate Professor with the School of Innovation and Entrepreneurship, Wuyi University. His current research interests include intelligence computing, text knowledge acquisition, big data, and intelligent recommendation.



YAN CUI is currently a Professor with the Faculty of Computer Science, Wuyi University. He is also the Dean of the School of Intelligent Manufacturing, Wuyi University and the China-Germany Artificial Intelligence Institute. His research interests include computer vision and computer graphic.



CHENCLIN DAI received the B.S. degree in computer science and technology (software engineering) from Lingnan Normal University, in 2020. He is currently pursuing the master's degree in electronic information from Wuyi University. His research interests include stereo matching and deep learning.