

Article

Multi-Scale Aggregation Stereo Matching Network Based on Dense Grouping Atrous Convolution

Qijie Zou *, Jie Zhang *, Shuang Chen, Bing Gao, Jing Qin and Aotian Dong

Information Engineering Faculty, Dalian University, Dalian 116622, China

* Correspondence: jessie_zou_zou@163.com (Q.Z.); zhangjietao666@163.com (J.Z.);
Tel.: +86-138-9842-6690 (Q.Z.); +86-178-6327-5822 (J.Z.)

Abstract: The key to image depth estimation is to accurately find corresponding points between the left and right images. A binocular camera can directly estimate the depth of the left and right image range, which completely avoids the requirement of target recognition accuracy in a monocular depth estimation. However, it is difficult for binocular stereo matching to accurately segment objects and find matching points in the ill-posed areas (weak texture, deformation, object edge, etc.) of the left and right images. In the semantic segmentation task, atrous convolution is used to solve the contradiction between the receptive field and the segmentation accuracy. This research focused on balancing the impact of holes on the segmentation task. In addition, in order to solve the issue where matching points in the ill-posed regions of left and right images are affected by noise, we used a 3D convolution to aggregate the cost volume to obtain better accuracy. However, the 3D convolution method is prone to mismatching in ill-posed areas of the image. To tackle the problems above, we proposed a dense grouping atrous convolution spatial pyramid pooling (DenseGASPP) method. The feature of the DenseGASPP method is that there is a dense connection between the group atrous convolutions to fully integrate feature information. This method can expand the receptive field and balance the effect of holes on the segmentation task. Moreover, we introduced multi-scale cost aggregation into our method, which uses the repeated exchange of information between cost volumes of different scales to obtain rich contextual information and reduce the mismatching of the network. To evaluate the performance of our method, we conducted several groups of typical algorithm experiments on the scene flow and KITTI 2015 standard datasets. From the results, our model achieves better performance, reducing the EPE from 1.09 to 0.67, which improves the mismatching ability of binocular depth estimation algorithm in ill-posed regions.

Keywords: binocular stereo matching; dense connection; atrous convolution; multi-scale cost aggregation



Citation: Zou, Q.; Zhang, J.; Chen, S.; Gao, B.; Qin, J.; Dong, A. Multi-Scale Aggregation Stereo Matching Network Based on Dense Grouping Atrous Convolution. *Appl. Sci.* **2023**, *13*, 7033. <https://doi.org/10.3390/app13127033>

Academic Editor: Andrea Prati

Received: 15 September 2022

Accepted: 30 September 2022

Published: 11 June 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Binocular stereo vision is an important branch of computer vision. It processes the real world by simulating the human vision system. In the 1980s, Marr [1] proposed a visual computing theory based on binocular matching, which laid the theoretical foundation for the development of binocular stereo vision. Because binocular vision can obtain rich target information and environment information, it is low cost and has the ability for stereo perception, and it is widely used in robot navigation [2], three-dimensional measurement [3], virtual reality, autonomous driving, and other fields.

Stereo matching is an essential part of binocular stereo vision and is mainly divided into traditional and deep learning methods. The traditional binocular stereo matching method relies on the experience of designers to select appropriate features, which has the defect of a low matching accuracy. With the development of CNN [4], the architecture has become an effective way to solve the problems in the binocular stereo matching algorithm. The end-to-end binocular stereo matching in the deep learning method uses CNN throughout the process; this seamlessly integrates all the steps in stereo matching,

inputting stereo images and directly outputting dense disparity maps [5]. Compared with the traditional method, it can complete a high-precision depth estimation from stereo image pairs. Therefore, the deep learning method is widely used.

End-to-end stereo matching has greatly improved the accuracy of binocular stereo matching, and this precise perception ability is urgently needed for autonomous driving. However, there are still challenges: one such challenge is improving the accuracy of stereo matching, including in the case of moving objects of different sizes. Therefore, it is urgent to design a more effective stereo matching network. The ASPP method [6] proposes using concatenate atrous convolution [7] layers with different dilation rates to obtain receptive fields of different sizes, which is conducive to multi-scale information [8,9]. The input image in autonomous driving scenes has a high resolution, which requires the neuron to have a larger receptive field. In order to obtain a sufficiently large receptive field in the ASPP method, a sufficiently large dilation rate must be used; however, as the dilation rate increases, the network gradually loses its modeling ability. Taking this into account, the DenseASPP [10] and GASPP [11] methods combine the advantages of the parallel and cascade use of void convolutional layers to obtain larger receptive fields and denser features, and thus improve the recognition ability of the algorithm when the target proportion changes. Although the above methods increase the receptive field to a certain extent, there is not enough attention paid to the contextual information.

Another difficulty is the matching accuracy present in challenging low-texture areas. The accuracy of stereo matching is affected by noise between pixels. In the traditional stereo matching method, the SGM [12] approach solves the influence of noise by finding the minimized matching cost energy function, but its energy function and solving process are manually designed, which limits the performance and generalization ability of the algorithm. The 3D CNN approach is able to directly extract features from the original input while simultaneously providing powerful regularization, which can be used in the cost aggregation step of stereo matching to eliminate noise between pixels. GC-Net [13] and PSM-Net [14] approaches both use stacked 3D CNNs for the regularization operations. However, when the image is in the ill-posed region, the traditional 3D convolution aggregation method cannot form a correct understanding of the details of the image, which affects the accuracy of the environment perception.

To solve the above issues, we propose a stereo matching network that is different from the previous methods. The network is composed of a dense grouping atrous spatial pyramid pooling (DenseGASPP) module and multi-scale cost aggregation module.

Our contributions are as follows:

- (1) We propose a dense grouping atrous convolution pyramid pooling (DenseGASPP) module to reduce the hole part of the network to obtain a large dense receptive field that improves the accuracy of segmentation.
- (2) We introduce multi-scale cost aggregation to replace the 3D convolution method commonly used in traditional stereo matching to reduce the mismatching problems in the ill-posed region.

DenseGASPP densely connects feature maps of different scales and balances the influence of network holes on the segmentation task while meeting the size change of moving objects, improving the accuracy of stereo matching. The multi-scale cost aggregation method we introduced can effectively utilize the details of the image and reduce the mismatching problem in the ill-posed region. Therefore, the network is suitable for the vision-based environment perception task used in autonomous driving.

2. Related Work and Methods

2.1. Related Work

2.1.1. End-to-End Module for Binocular Stereo Matching

Conventional stereo matching can be roughly divided into global and local methods [15]. The global method usually solves an optimization problem by minimizing a global objective function that includes data and smoothness terms [16,17]; the local method

only considers neighbor information [18,19], which is faster than the global method [15,20]. Although traditional methods have made great progress, they are still affected by challenging situations [21].

Inspired by FlowNet [22], the DispNet approach proposed by Mayer et al. [5] was the first end-to-end convolutional neural stereo matching network. The input binocular stereo are rectified image pairs, and the predicted disparity map is directly output through the designed convolution neural network model. Compared with the traditional methods, DispNet runs faster and has higher accuracy. Compared with the real-time algorithm in [23], the error is reduced by 30%. PSM-Net proposes an end-to-end learning framework for stereo matching without any post-processing, which improves the accuracy by 20% compared with GC-Net. Although end-to-end binocular stereo matching is the focus of the current research, there are still many defects in this method. Because binocular stereo matching aims to obtain depth information by calculating the disparity of two similar pixels [24], the intuitive performance is that the more obvious the feature changes are, the easier it is to find the similarity, so that the accuracy is higher in the region with obvious feature changes. This inherent characteristic also causes the end-to-end stereo matching method to have weak textures, deformation, and object edges prone to error matching [25]. Therefore, this paper explores and proposes solutions to these deficiencies.

To solve the above issue, this paper proposes fusing multi-resolution features in the feature extraction stage to enrich the feature information of complex regions and to improve the existing matching cost volume model. The matching cost volume is composed of multi-scale features and scale-based inner products to describe the similarity between features. In the cost aggregation part, multi-scale cost aggregation [26] is designed to further the fusion of contextual information for rich details [21].

2.1.2. Atrous Convolution

Atrous convolution is a method of increasing the receptive field without increasing the computation or reducing the resolution of the feature map [10,11]. In atrous convolution, the receptive field refers to the size of the area mapped from the original image by pixels in the feature map output by each layer of the convolution neural network [10], and the size of the receptive field is closely related to the dilation rate. In order to increase the receptive field and reduce the calculation in a deep network, it is always necessary to carry out down-sampling and pooling operations, which can increase the receptive field but reduce the spatial resolution. Therefore, if you wish to increase the receptive field while maintaining the resolution, you can use atrous convolution. By this method, the receptive field and resolution can be balanced, and multi-scale information can be obtained by adjusting the dilation rate.

The similarity between atrous convolution and ordinary convolution is that the size of convolution kernel is the same, and the number of parameters in the neural network remains unchanged. The difference is that atrous convolution preserves the original image information while increasing the receptive field. The structure is shown in Figure 1.

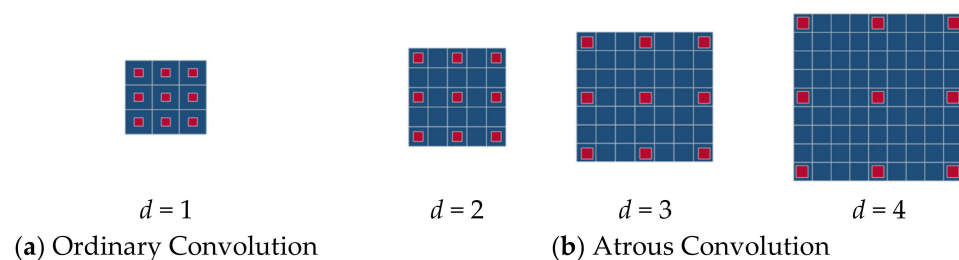


Figure 1. The comparison of ordinary convolution and atrous convolution: (a) schematic diagram of ordinary convolution; (b) schematic diagram of atrous convolution. The interval between adjacent convolution kernels is $d - 1$.

The receptive field size of a convolution kernel can be formulated using the following equation:

$$R = (d - 1) \times (K - 1) + K \quad (1)$$

where d represents the dilation rate and K represents the size of a kernel. R is the size of the receptive field.

Obviously, atrous convolution increases the receptive field without losing the size of the feature map, but it has an inherent problem known as the grid effect. In the result of a certain layer of atrous convolution, adjacent pixels are obtained from the convolution of mutually independent subsets, and the lack of dependence on each other leads to the loss of local information, which is harmful to the segmentation of small objects [27]. In order to solve this problem of atrous convolution, the ASPP method proposes connecting the feature maps generated by hole convolution under different dilation rates so that the neurons in the output feature map contain multiple acceptance domain sizes, encode multi-scale information, and finally improve the matching accuracy. However, with the increase in dilation rate, atrous convolution becomes more and more ineffective and gradually loses its modeling ability. Therefore, it is very essential to design a network structure that can encode multi-scale information and obtain a large receptive field. In order to obtain a large enough receptive field and avoid convolution degradation, Yang et al. [10] proposed DenseASPP, which combines the advantages of the parallel and cascading usage of the atrous convolution layer, which obtains a larger receptive field and denser features, along with improving the recognition ability of the algorithm when the target ratio changes. This leads to an improvement of the robustness of the matching.

Regarding the DenseASPP method, stacking the two convolution layers together can provide a larger receptive field. Suppose we have two convolution layers. The convolution kernel size is K_1 and K_2 respectively, and the new receptive field R is calculated as follows:

$$\begin{cases} R_1 = (d_1 - 1) \times (K_1 - 1) + K_1 \\ R_2 = (d_2 - 1) \times (K_2 - 1) + K_2 \\ R = R_1 + R_2 - 1 \end{cases} \quad (2)$$

where R_1, d_1 are the receptive field and dilation rate of the first convolution layer, respectively, and R_2, d_2 are the receptive field and dilation rate of the second convolution layer, respectively.

Although DenseASPP increases the receptive field to some extent, it still has defects in enhancing the correlation between contextual information. Inspired by DenseNet [28], we in part connected the initial feature maps obtained from the feature extraction in a dense connection approach so that the feature information contained in the different feature maps could complement each other, which is conducive to enhancing the correlation between information and improving the accuracy of network.

2.1.3. Cost Aggregation

Cost aggregation is an essential module in stereo matching and is used to solve the issue where pixel matching is affected by noise; the result is that the cost value after aggregation can more accurately reflect the correlation between pixels.

The feature extraction of GC-Net is only completed in some stacked convolution layers [29], and the ill-posed region is not taken into account when adjusting the network matching accuracy in the cost aggregation part. As a result, it is difficult to form an understanding of the weak texture, deformation, and object edge.

PSM-Net introduces the pyramid pooling module to incorporate global context information into image features, significantly improving the cost aggregation compared with GC-Net. PSM-Net provides a stacked hourglass 3D CNN to adjust the matching cost volume from fine to coarse and then from coarse to fine; this is performed to expand the regional support of the context information in the cost volume stage [14] so that the stereo

matching process achieves high accuracy. However, this method still has the problem of mismatching in the low texture, deformation, and object edge regions.

The features in the cost aggregation stage need to contain spatial detailed information. In particular, more information needs to be sampled in the ill-posed area. Because the image feature resolutions obtained at different scales are different, inspired by [21,26], we constructed multi-scale cost volumes by performing feature correlation operations on feature maps at different resolutions and fused the cost volumes at different scales. In this way, each representation, from high resolution to low resolution, repeatedly receives information from other representations to obtain rich high-resolution representations, which can have high matching accuracy in the weak texture, deformation, and object edge regions.

2.2. Methods

2.2.1. Network Architecture

End-to-end binocular stereo matching consists of feature extraction, cost volume construction, cost aggregation, and disparity regression stages. We propose a network model, which is composed of a DenseGASPP and multi-scale cost aggregation module for feature extraction and cost aggregation, respectively. The architecture is shown in Figure 2.

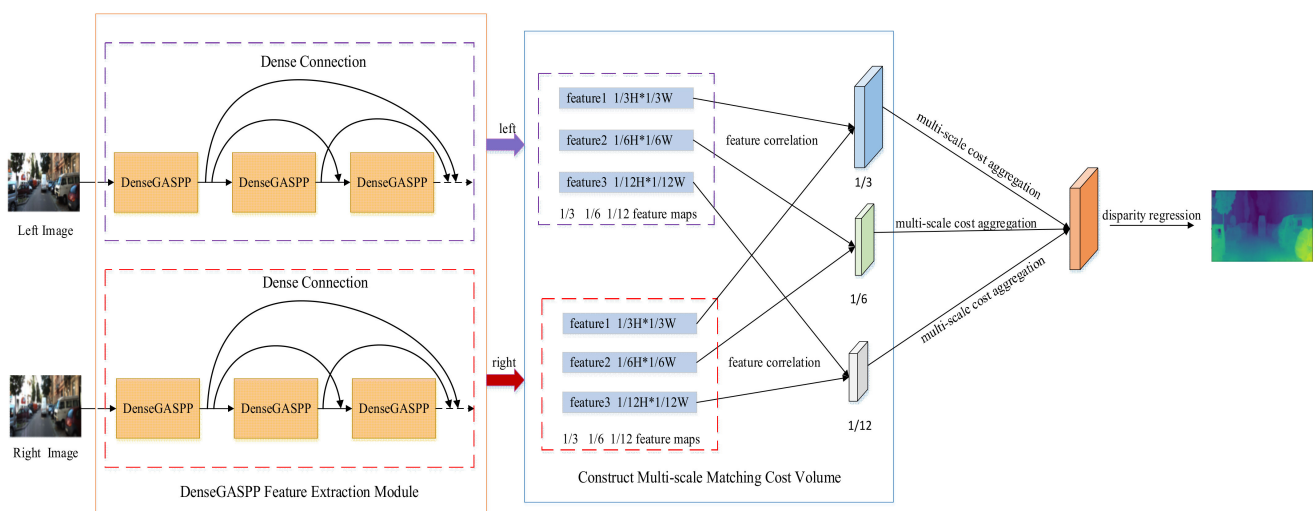


Figure 2. The architecture of the multi-scale aggregation network based on dense grouping atrous convolution.

DenseGASPP utilizes dense grouping atrous convolution to extract features of the left and right images and acquire feature maps with different resolutions.

For the cost volume part, feature correlation is used to correlate the left and right features of feature maps with different resolutions to construct differently scaled cost volumes.

Multi-scale cost aggregation utilizes intra-scale cost aggregation and cross-scale cost aggregation to aggregate features between the cost volumes of different scales.

Disparity regression utilizes the soft-argmin function [13].

In the above four parts, our work focused on feature extraction and cost aggregation.

2.2.2. DenseGASPP for Dense Feature Extraction

Feature extraction is the premise of correctly estimating the disparity in the stereo matching task. Making full use of global context information in feature extraction is helpful for capturing different proportions of objects and improving the accuracy of object segmentation, which is essential for stereo matching.

DenseGASPP was utilized for feature extraction. In the DenseGASPP module, we designed three atrous convolution layers with a dense connection, allocated two convolution layers with a continuous small dilation rate in each layer, connected the two convolution

layers with a continuous dilation rate in each layer, and lastly connected the feature maps of three resolutions obtained from the three layers of atrous convolution layers to fully fuse the feature information. This method can balance the influence of the hole part on the segmentation task. The model is shown in Figure 3. The fusion process is shown in Equation (3).

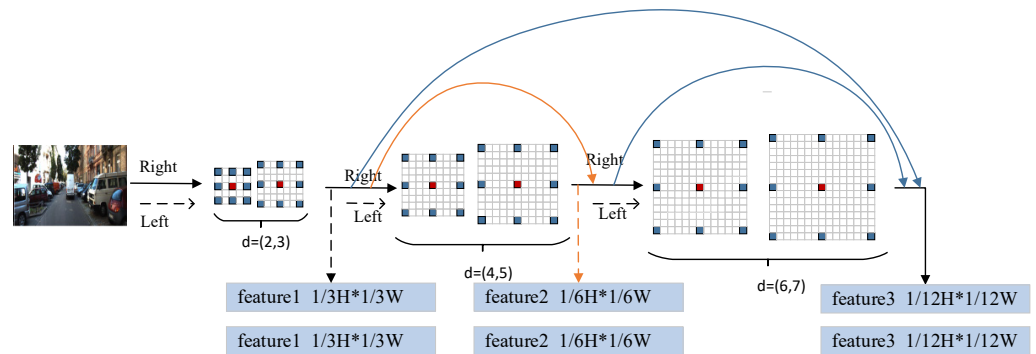


Figure 3. The structure of the DenseGASPP module. We used the right sub-graph as an example to introduce the DenseGASPP module. Three groups of atrous convolution layers with the following dilation rates $\{(2,3) (4,5) (6,7)\}$ are designed in the DenseGASPP module, and two groups of convolution layers with a continuous small disparity rate are distributed in each group. In addition, the results between the three groups of atrous convolution layers are densely connected, that is, the feature maps between the groups are concatenated, which is helpful for obtaining more feature information.

DenseGASPP can be written as:

$$\begin{cases} X_i = G(X_{i-1}), & i = 1 \\ X_i = [U_{X_{i-1}}, G(X_{i-1})], & i = 2, 3 \end{cases} \quad (3)$$

where i represents the number of network layers, X_i represents the output of dense grouping atrous convolution at layer i , X_{i-1} represents the network input, G represents the grouping atrous convolution function (which is a combined operation), U represents the up-sampling operation, and $[U_{X_{i-1}}, G(X_{i-1})]$ represents the concatenation, that is, the combination between different network layers.

In general, DenseGASPP is divided into two steps for feature extraction. First, feature maps of different scales containing rich context information are obtained using grouping atrous convolution at each layer. Then, different scale feature maps are concatenated to maximize the use of object detail information to improve the matching accuracy of the network for all objects, especially small objects.

2.2.3. Multi-Scale Cost Aggregation for Reduced Mismatching

The fundamental purpose of cost aggregation is to make the cost accurately reflect the correlation between pixels and solve the issue of pixel matching being affected by noise. In fact, cost aggregation is similar to the disparity propagation step. Regions with a high signal-to-noise ratio have a good matching effect, but only local correlation is considered when calculating the initial cost. As a result, the disparity is not the smallest in the low signal-to-noise ratio and ill-posed regions of the image, so it is difficult to find the correct homonymous points. By propagating to these regions through cost aggregation, the optimal disparity can be more accurately acquired. Finally, the cost volume of all images can accurately reflect the true correlation.

In the ill-posed area of the image, there is less information, so it is necessary to obtain the identifiable details from the high-resolution features so as to accurately find the homonymous point. We used cost aggregation at different scales, which can acquire detailed information and rich contextual information, which helps to reduce the problem of mismatching in ill-posed regions.

For the multi-scale cost aggregation, we introduced the intra-scale and cross-scale cost aggregation. The model is shown in Figure 4. Because the feature resolution is the same at the same scale, intra-scale cost aggregation only aggregates the cost volumes with the same resolution, and the copy operation is directly adopted between the cost volumes. Cross-scale cost aggregation is used to aggregate the characteristics between cost volumes of different scales [21]. The cross-scale aggregation operation can be divided into two cases: (1) up-sampling: the scale of the current cost volume is larger than that of other cost volumes. This entails the up-sampling of other cost volumes to the current scale to align the resolution; (2) down-sampling: the scale of the current cost volume is smaller than that of other cost volumes. This entails the down-sampling other cost volumes to the current scale to align the resolution.

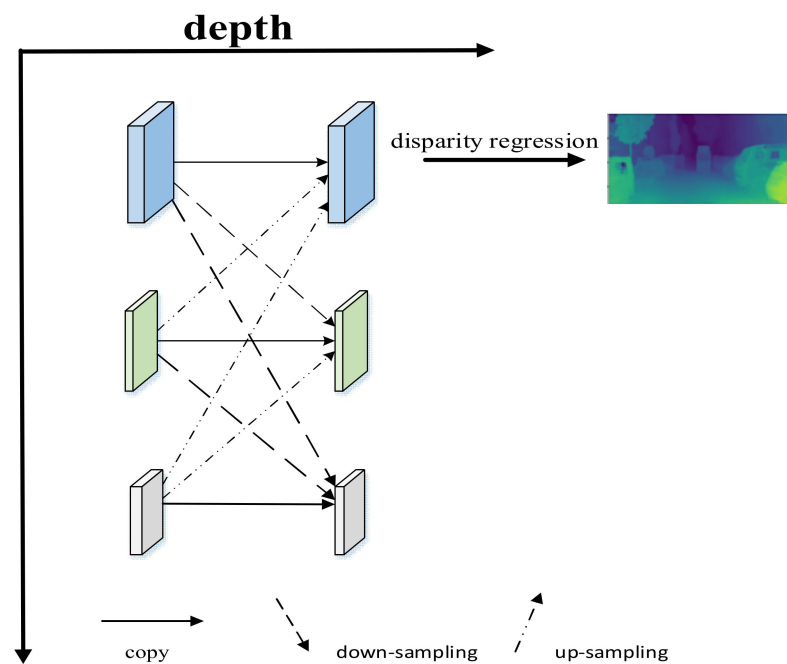


Figure 4. Schematic diagram of the multi-scale cost aggregation module.

Starting from the high-resolution subnet as the first stage, we gradually increased the high-resolution to low-resolution subnets to form more stages and connected the multi-resolution subnets in parallel. In the whole process, we performed multi-scale repeated fusion by repeatedly exchanging the information on parallel multi-resolution subnetworks. Finally, the high-resolution features were obtained.

2.2.4. Disparity Regression

Disparity regression was proposed to estimate the continuous disparity maps.

For each pixel, we used the soft-argmin function to obtain disparity prediction $\tilde{d}$:

$$\tilde{d} = \sum_{d=0}^{D_{max}-1} d \times \sigma(c_d), \quad (4)$$

where D_{max} is the maximum disparity range, σ is the softmax function, and c_d is the aggregate matching cost of disparity candidate d . $\sigma(c_d)$ can be regarded as the probability of disparity d .

2.2.5. Loss Function

DenseGASPP, we propose, is trained end-to-end and supervised by the ground truth disparity. However, for the KITTI dataset, the high sparsity of the disparity ground truth may not actively promote our learning process. Inspired by the knowledge reported in [30], we used the prediction results of the pre-trained stereo model as the pseudo ground truth

supervision. Specifically, we used a pre-trained model to predict the disparity maps on the training set and used the prediction results as a pseudo label. In other words, there is no real ground truth disparity in the pixels.

For the disparity prediction $\tilde{d}_i, i = 1, 2, 3$, bilinear up-sampling was first performed to the original resolution. The corresponding loss function is defined as:

$$L_i = \sum_p V(p) \cdot \mathcal{L}(\tilde{d}_i(p), D_{gt}(p)) + (1 - V(p)) \cdot \mathcal{L}(\tilde{d}_i(p), D_{pseudo}(p)), \quad (5)$$

where $V(p)$ is a binary mask to denote whether the ground truth disparity for pixel p is available, $\mathcal{L}$ is the smooth L_1 loss [9], D_{gt} is the ground truth disparity, D_{pseudo} is the pseudo ground truth, and i represents the sequence of scales.

The final loss function is the combination of all disparity predicted losses:

$$L = \sum_{i=1}^N \lambda_i \cdot L_i \quad (6)$$

where λ_i is a scalar for balancing the different terms.

3. Experiments

In this section, the relevant experiment settings and results are described. We evaluated the key components of our network under the Scene Flow [5] and KITTI 2015 [31] datasets. In addition, we also compared our method with the state-of-the-art stereo matching methods using the KITTI benchmark.

3.1. Dataset

Scene flow and KITTI 2015 datasets were utilized in our network implementation process. For the scene flow dataset, we used all of the training sets (35,454 stereo pairs) to train and evaluate the standard test sets (4370 stereo pairs), with the original images being cropped to 288×512 as inputs. For the KITTI 2015 dataset, 1000 epochs of benchmark tests were performed using the model pre-trained on the scene flow dataset, with an initial learning rate of 0.0001.

Scene flow: a large synthetic dataset, including FlyingThings3D and Driving and Monkaa. The dataset contains binocular stereo image pairs and ground truth. There are 39,824 sets of data, of which 35,454 are training sets and 4370 are testing sets. The image resolution is 960×540 .

KITTI 2015: a dataset collected from a real street scene, including 200 pairs of stereo images for training and 200 pairs of stereo images for testing. The sparse disparity image collected by LiDAR remote sensing is provided as the ground truth image. The resolution of the dataset is 1240×376 .

3.2. Evaluation Metrics

Because the essence of binocular stereo matching is to find the homonymous points of the left and right images, the error directly affects the accuracy of the stereo matching. Thus, the end-point error (EPE) and 1-pixel error are used to represent the accuracy of the matching. The calculation formula of the EPE is as follows:

$$EPE = \frac{1}{N} \sum_{m \in N} \sqrt{(d_m - \hat{d}_m)^2} \quad (7)$$

where N represents the total number of pixels, d_m represents the ground truth at the pixel m , and $\hat{d}_m$ represents the predicted disparity value at the pixel m .

The EPE is the average disparity error in pixels, and the error is inversely proportional to the matching accuracy. The smaller the EPE, the higher the matching accuracy. The 1-pixel error is the average percentage of pixels that have an EPE bigger than 1 pixel.

In addition, the real-time nature of the network requires special attention in the field of automatic driving, and its data processing must meet the constraints of driving speed

and safe distance. We used the length of inference time to reflect the speed of the network modeling. The inference time is inversely proportional to the modeling speed. The longer the inference time is, the slower the modeling speed is. The shorter the inference time is, the faster the modeling speed is.

3.3. Implementation Details

Using Ubuntu 20.04, we used the PyTorch [32] deep learning framework to implement our method; furthermore, we used Adam [33] ($\beta_1 = 0.9$, $\beta_2 = 0.999$) as the optimizer. The network was trained for 128 epochs on the scene flow dataset. The batch size was set to 16. The learning rate started from 0.001 and decreased by half every 10 epochs after the 20 epoch mark. The loss weights were set from high to low: $\lambda_1 = \lambda_2 = \lambda_3 = 1.0$, $\lambda_4 = 2/3$, $\lambda_5 = 1/3$. The parameter configuration of the experiment is shown in Table 1.

Table 1. The parameter configuration of the experiment.

Software/Hardware	Configuration
CPU	Intel Core i9-10920x
GPU	RTX 6000
Operating system	Ubuntu 20.04
CUDA version	CUDA 10.0
Language	Python 3.7
Deep learning framework	PyTorch 1.3.0

3.4. Experimental Results

3.4.1. Performance Analysis of Stereo Matching

Due to the diversity and complexity of road environment and weather environment, it is easy to be affected by many uncertain factors in the modeling process of autonomous driving. Therefore, these problems should be paid attention to in the process of stereo matching. We compare with the state-of-the-art network on scene flow. As shown in Table 2.

Table 2. Comparison of our DenseGASPP network with the state-of-the-art stereo matching networks under the scene flow dataset.

Network Model	EPE	>1 px
PSM-Net	1.09	12.1
GA-Net	0.87	9.9
AANet	0.86	9.3
DenseGASPP	0.68	7.2

As can be seen from the Table 2, DenseGASPP had the smallest EPE and 1-pixel error with values of 0.67 and 7.2, respectively, which was better than the state-of-the-art networks.

In order to achieve the best performance of the network, we set up six groups of experiments on the dilation rate and dilation rate interval of the DenseGASPP module when the cost volume scale was fixed at $\{(1/2), (1/4), (1/8)\}$. When the dilation was certain, we conducted three groups of experiments in the cost volume stage. The experimental results are shown in Tables 3 and 4, respectively.

As can be seen from Table 3, when the dilation rate interval was one and the dilation rate combination was $\{(2,3), (4,5), (6,7)\}$, the EPE on scene flow was 0.86 and the EPE on KITTI 2015 was 0.69, and the network could achieve the best performance. As can be seen from Table 4, when the cost volume scale was $\{1/3, 1/6, 1/12\}$, the EPE on scene flow was 0.67 and the EPE on KITTI 2015 was 0.44, and the network could achieve the best performance.

Table 3. Experimental evaluation of different dilation rates of the DenseGASPP module.

Serial Number	Interval of Dilation Rate	Group 1	Dilation Group Group 2	Group 3	Scene Flow EPE	KITTI 2015 EPE
1	1	(2,3)	(4,5)	(6,7)	0.86	0.69
2	2	(2,3)	(5,6)	(8,9)	0.93	0.77
3	2	(3,4)	(6,7)	(9,10)	0.92	0.75
4	3	(2,3)	(6,7)	(10,11)	0.98	0.71
5	4	(2,3)	(7,8)	(12,13)	1.02	0.94
6	5	(2,3)	(8,9)	(14,15)	1.14	0.96

Table 4. Experimental evaluation of the cost volume in different scales.

Dilation Rate	Scale of Cost Volume	Scene Flow EPE	KITTI 2015 EPE
(2,3) (4,5) (6,7)	1/2 1/4 1/8	0.86	0.69
	1/3 1/6 1/12	0.67	0.44
	1/4 1/8 1/16	0.92	0.74

Based on the experimental results in Table 3 and 4, we selected the dilation rate $\{(2,3), (4,5), (6,7)\}$, cost volume scale $\{1/3, 1/6, 1/12\}$ that should be used as the parameters of the DenseGASPP and multi-scale cost volume, respectively.

3.4.2. Accuracy Analysis of Disparity Map

For intelligent driving tasks, we chose to evaluate the disparity map on the KITTI 2015 dataset with street scenes. Figure 5 shows the disparity map effect of PSM-Net, GWC-Net, and DenseGASPP. These results were reported by the KITTI evaluation server.

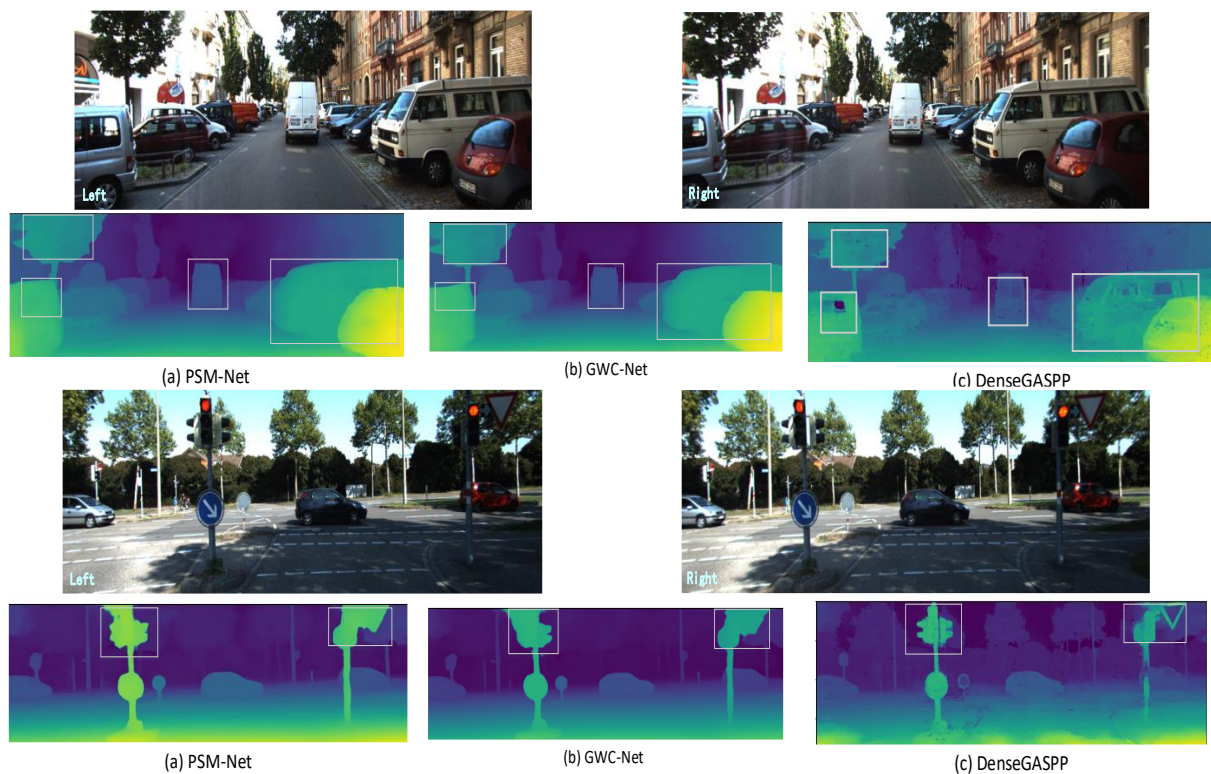


Figure 5. Disparity map effect under the KITTI 2015 dataset. The first and third rows are the input image of the stereo image pairs, (a) PSM-Net, (b) GWC-Net, (c) DenseGASPP are the disparity maps obtained in the network for the input pairs.

As shown by the regions we selected with white boxes, our network showed a more accurate environmental perception in both the weak texture area of the object and the object edge compared with the state-of-the-art networks.

3.5. Ablation Study

3.5.1. Ablation Study of DenseGASPP Module

In this section, we tested the impact of the difference of feature extraction networks on network performance. In autonomous driving, the real-time performance of binocular vision is particularly important. Real-time performance means that the data processing of the visual perception system must meet the constraints of driving speed and safe distance. We carried out a validation using the scene flow dataset of multiple scenes and KITTI 2015 dataset of street scenes, respectively. We used DenseGASPP instead of the feature extraction part in the traditional method and used the suffix D to represent the obtained model. The results are shown in Table 5.

Table 5. Ablation study results of the DenseGASPP module.

Network Model	Inference Time (s)	Scene Flow EPE	KITTI 2015 EPE
PSM-Net	0.047	0.97	0.75
PSM-Net-D	0.053	0.85	0.62
AANet	0.095	0.87	0.68
AANet-D	0.051	0.67	0.44
GASPP	0.049	0.92	0.73
GASPP-D	0.055	0.81	0.63

As observed from the experimental results in Table 5, our method had the highest accuracy, but the reasoning speed was slower than the SPP and GASPP methods. However, considering that vehicles are driving in the living area, the traffic speed limit ranges from 30 km/h to 60 km/h. If the driving speed of the vehicle is 60 km/h, the maximum braking distance required is 19 m. The time required to process the command data before the braking operation mainly includes the detection and system response time. Take our longest network detection time of 0.055 s as an example. According to the requirements of autonomous driving product standards, the response time of the electronic control system is 0.1 s. Within this 0.155 s, the vehicle will travel at a constant speed of 2.58 m, and the sum of the braking distance and constant speed travel distance will arrive at the final stopping distance of 21.58 m. Taking Tesla's vision sensor as an example, its front-view narrow field camera can monitor a distance of up to 250 m, and the front-view main field and wide field cameras can detect 150 m and 60 m, respectively. The maximum monitoring distance of the side front-view camera is 80 m, and the maximum monitoring distance of the side rear-view camera is 100 m. The maximum monitoring distance of the rear-view camera is 50 m. By a comprehensive comparison, the model DenseGASPP proposed by us fully meets the constraints of driving speed and safe distance.

3.5.2. Ablation Study of Multi-Scale Cost Aggregation

In autonomous driving, the car drives in non-repetitive scenes. There are many complex scenes, so the modeling process is often affected by the image's ill-posed area. There will be mismatching in the binocular stereo matching process.

In order to verify the effectiveness of the multi-scale cost aggregation in reducing the image mismatching in the ill-posed area, we conducted validation experiments on the scene flow dataset with multiple scenes, and compared the three state-of-the-art network models. We used multi-scale cost aggregation to replace the 3D convolution method in the cost aggregation stage. The suffix S to represent the obtained model. The experimental results are shown in Table 6.

Table 6. Ablation study results of multi-scale cost aggregation under the scene flow dataset.

Network Model	EPE	>1 px
PSM-Net	1.09	10.3
PSM-Net-S	0.97	10.2
GC-Net	2.51	16.9
GC-Net-S	0.98	10.8
GA-Net	0.87	9.9
GA-Net-S	0.88	9.3

By comparing the EPEs in Table 6, we found that our method has a smaller EPE and higher accuracy than the state-of-the-art methods.

4. Conclusions

We propose a dense grouping atrous convolution spatial pyramid pooling module. This module uses the atrous convolution of groups and dense connection between convolution layers to form a large receptive field and a feature map containing multi-scale information, which effectively enhances the network's understanding of global information and improves the accuracy of stereo matching. In addition, the cost aggregation is divided into two parts: intra-scale cost aggregation and cross-scale cost aggregation. Through the continuous fusion of information on different scales and through the alignment resolution, we can extract the features with a good recognition degree and effectively reduce the false matching rate of the algorithm in the ill-posed regions of the image.

For the depth estimation task in autonomous driving, the experiment shows that the method can achieve high accuracy on the scene flow and KITTI 2015 datasets and meets the requirements of the autonomous driving task. Our method is an end-to-end stereo matching method, and most end-to-end stereo matching methods are based on datasets with ground truth, that is, supervised methods. The unsupervised stereo matching method does not rely on ground truth and has a wider range of applications, which will be our focus in subsequent research. Because the dataset needs to be labeled as a supervised method, it requires high labor cost. In addition, there are various scenes in the urban environment, and it is not realistic to comprehensively annotate the collected images. Unsupervised methods can avoid these constraints.

Author Contributions: Conceptualization, Q.Z. and J.Z.; methodology, J.Z.; software, J.Z.; validation, J.Z. and A.D.; formal analysis, Q.Z.; investigation, J.Z.; resources, Q.Z.; data curation, J.Z.; writing—original draft preparation, J.Z.; writing—review and editing, Q.Z.; visualization, J.Z.; supervision, B.G.; project administration, J.Q.; funding acquisition, S.C. All authors have read and agreed to the published version of the manuscript.

Funding: The research is supported by the National Natural Science Foundation of China (Grant No. 11701061): Research on SA algorithm for nonconvex stochastic semidefinite programming, and the scientific research fund project of Liaoning Province (Project No. LJKZ1180).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Marr, D. *Vision a Computational Investigation into the Human Representation and Processing of Visual Information*; MIT Press: London, UK, 1983; Volume 8.
2. Xiu-Juan, L.I.; Liu, W.; Shan-Hong, L.I. Robust Control Algorithm of Bionic Robot Based on Binocular Vision Navigation. *Comput. Sci.* **2017**, *21*, 318–322.

3. Trzcinski, T.; Christoudias, M.; Fua, P.; Lepetit, V. Boosting Binary Keypoint Descriptors. *Computer Vision and Pattern Recognition*. In Proceedings of the CVPR–2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; IEEE: Piscataway, NJ, USA, 2013.
4. LeCun, Y.; Boser, B.; Denker, J.S.; Henderson, D.; Howard, R.E.; Hubbard, W.; Jackel, L.D. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Comput.* **1989**, *1*, 541–551. [\[CrossRef\]](#)
5. Mayer, N.; Ilg, E.; Hausser, P.; Fischer, P.; Cremers, D.; Dosovitskiy, A.; Brox, T. A Large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 4040–4048.
6. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848. [\[CrossRef\]](#) [\[PubMed\]](#)
7. Yu, F.; Koltun, V.; Funkhouser, T. Dilated Residual Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
8. Zhu, Z.; He, M.; Dai, Y.; Rao, Z.; Li, B. Multi-scale cross-form pyramid network for stereo matching. In Proceedings of the 2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA), Xi'an, China, 19–21 June 2019; IEEE: Piscataway, NJ, USA, 2019.
9. Zhu, Z.; Guo, W.; Chen, W.; Li, Q.; Zhao, Y. MPANet: Multi-Scale Pyramid Aggregation Network For Stereo Matching. In Proceedings of the 2021 IEEE International Conference on Image Processing (ICIP), Virtual, 19–22 September 2021; IEEE: Piscataway, NJ, USA, 2021.
10. Yang, M.; Yu, K.; Zhang, C.; Li, Z.; Yang, K. DenseASPP for Semantic Segmentation in Street Scenes. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; IEEE: Piscataway, NJ, USA, 2018.
11. Zou, Q.; Yu, J.; Fang, H.; Qin, J.; Zhang, J.; Liu, S. Group-Based Atrous Convolution Stereo Matching Network. *Wirel. Commun. Mob. Comput.* **2021**, *2021*, 7386280. [\[CrossRef\]](#)
12. Yang, P.; Sun, X.; Li, W.; Ma, S.; Wu, W.; Wang, H. SGM: Sequence Generation Model for Multi-label Classification. In Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, NM, USA, 21–25 August 2018.
13. Kendall, A.; Martirosyan, H.; Dasgupta, S.; Henry, P.; Kennedy, R.; Bachrach, A.; Bry, A. End-to-end learning of geometry and context for deep stereo regression. In Proceedings of the IEEE International Conference on Computer Vision 2017, Venice, Italy, 22–29 October 2017; pp. 66–75.
14. Chang, J.; Chen, Y. Pyramid stereo matching network. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5410–5418.
15. Scharstein, D.; Szeliski, R. A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms. *Int. J. Comput. Vis.* **2002**, *47*, 7–42. [\[CrossRef\]](#)
16. Sun, J.; Zheng, N.; Shum, H. Stereo matching using belief propagation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2003**, *7*, 787–800.
17. Kolmogorov, V.; Zabih, R. Computing visual correspondence with occlusions using graph cuts. In Proceedings of the Eighth IEEE International Conference on Computer Vision, Vancouver, BC, Canada, 7–14 July 2001; Volume 2, pp. 508–515.
18. Yoon, K.; Kweon, I.S. Adaptive support-weight approach for correspondence search. *IEEE Trans. Pattern Anal. Mach. Intell.* **2006**, *4*, 650–656. [\[CrossRef\]](#) [\[PubMed\]](#)
19. Hosni, A.; Rhemann, C.; Bleyer, M.; Rother, C.; Gelautz, M. Fast Cost-Volume Filtering for Visual Correspondence and Beyond. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *35*, 504–511. [\[CrossRef\]](#) [\[PubMed\]](#)
20. Min, D.; Lu, J.; Do, M.N. A revisit to cost aggregation in stereo matching: How far can we reduce its computational redundancy? In Proceedings of the 2011 International Conference on Computer Vision, Washington, DC, USA, 6–13 November 2011; pp. 1567–1574.
21. Xu, H.; Zhang, J. AANet: Adaptive Aggregation Network for Efficient Stereo Matching. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; IEEE: Piscataway, NJ, USA, 2020.
22. Dosovitskiy, A.; Fischer, P.; Ilg, E.; Hausser, P.; Hazirbas, C.; Golkov, V.; van der Smagt, P.; Cremers, D.; Brox, T. FlowNet: Learning optical flow with convolutional networks. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 2758–2766.
23. Chang, Q.; Maruyama, T. Real-Time Stereo Vision System: A Multi-Block Matching on GPU. *IEEE Access* **2018**, *6*, 42030–42046. [\[CrossRef\]](#)
24. Wang, D.; Hua, L.; Cheng, X. A Miniature Binocular Endoscope with Local Feature Matching and Stereo Matching for 3D Measurement and 3D Reconstruction. *Sensors* **2018**, *18*, 2243. [\[CrossRef\]](#) [\[PubMed\]](#)
25. Kun, Z.; Xiangxi, M.; Cheng, B. Review of stereo matching algorithms based on deep learning. *Comput. Intell. Neurosci.* **2020**, *14*, 8562323.
26. Sun, K.; Xiao, B.; Liu, D.; Wang, J. Deep high-resolution representation learning for human pose estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–17 June 2019; pp. 5693–5703.
27. Yu, F.; Koltun, V. Multi-Scale Context Aggregation by Dilated Convolutions. In Proceedings of the International Conference on Learning Representations—ICLR, San Juan, Puerto Rico, 2–4 May 2016.

28. Huang, G.; Liu, Z.; Van der Maaten, L.; Weinberger, Q.K. Densely connected convolutional Networks. In Proceedings of the Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
29. Calonder, M.; Lepetit, V.; Strecha, C.; Fua, P. BRIEF: Binary robust independent elementary features. In Proceedings of the European Conference on Computer Vision (ECCV), Heraklion, Greece, 5–11 September 2010; pp. 778–792.
30. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *arXiv* **2015**, arXiv:1503.02531.
31. Object scene flow for autonomous vehicles. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3061–3070.
32. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. Pytorch: An imperative style, high-performance deep learning library. In Proceedings of the Advances in Neural Information Processing Systems: Annual Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; pp. 8024–8035.
33. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.