Mathematical Biosciences and Engineering

MBE, 17(6): 7787–7803.
DOI: 10.3934/mbe.2020396
Received: 28 July 2020
Accepted: 19 October 2020
Published: 06 November 2020

http://www.aimspress.com/journal/MBE

*Research article*

# An end-to-end stereo matching algorithm based on improved convolutional neural network

**Yan Liu*, Bingxue Lv, Yuheng Wang and Wei Huang**

College of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou 45000, China

*** Correspondence:** Email: lyanzju@163.com.

**Abstract:** Deep end-to-end learning based stereo matching methods have achieved great success as witnessed by the leaderboards across different benchmarking datasets. Depth information in stereo vision systems are obtained by a dense and accurate disparity map, which is computed by a robust stereo matching algorithm. However, previous works adopt network layer with the same size to train the feature parameters and get an unsatisfactory efficiency, which cannot be satisfied for the real scenarios by existing methods. In this paper, we present an end-to-end stereo matching algorithm based on "downsize" convolutional neural network (CNN) for autonomous driving scenarios. Firstly, the road images are feed into the designed CNN to get the depth information. And then the "downsize" full-connection layer combined with subsequent network optimization is employed to improve the accuracy of the algorithm. Finally, the improved loss function is utilized to approximate the similarity of positive and negative samples in a more relaxed constraint to improve the matching effect of the output. The loss function error of the proposed method for KITTI 2012 and KITTI 2015 datasets are reduced to 2.62 and 3.26% respectively, which also reduces the runtime of the proposed algorithm. Experimental results illustrate that the proposed end-to-end algorithm can obtain a dense disparity map and the corresponding depth information can be used for the binocular vision system in autonomous driving scenarios. In addition，our method also achieves better performance when the size of the network is compressed compared with previous methods.

**Keywords:** image sensor; stereo matching; binocular vision; convolutional neural network

## 1. Introduction

Deep learning is the most useful tool for many applications, such as parameters compressing [1], sentiment analysis [2], information security [3], person re-identification [4], compressive sensing [5,6], object tracking [7], image classification [8–10], etc.. Nowadays, image sensors are widely used in the fields such as robotics [11], automatic-driving [12], medical diagnosis [13], security monitor [14], Augmented Reality (AR) [15], which are the core components of vision systems. Stereo matching aims at estimating the disparity map between a rectified image pair, which is of great importance to various applications such as obstacle avoidance for robot navigation [16,17], 3D scene reconstruction for augmented and virtual reality system [18], and 3D visual object tracking and location [19,20]. Depth information captured from image sensors can be calculated by the following typical structure: Time of Flight (ToF) [21,22], structured light [23,24], and binocular vision [25,26], in which ToF and structured light have high accuracy in a particular scene, but they need high implementation cost. In the field of automatic-driving [12], binocular stereo vision technology [27] is widely adopted due to the advantages of low cost, information abundant, and high robustness in object recognition, which can extract the dense parallax map and achieves the segmentation of road lane lines. The binocular vision system architecture used in the autonomous driving scenario is shown in Figure 1.

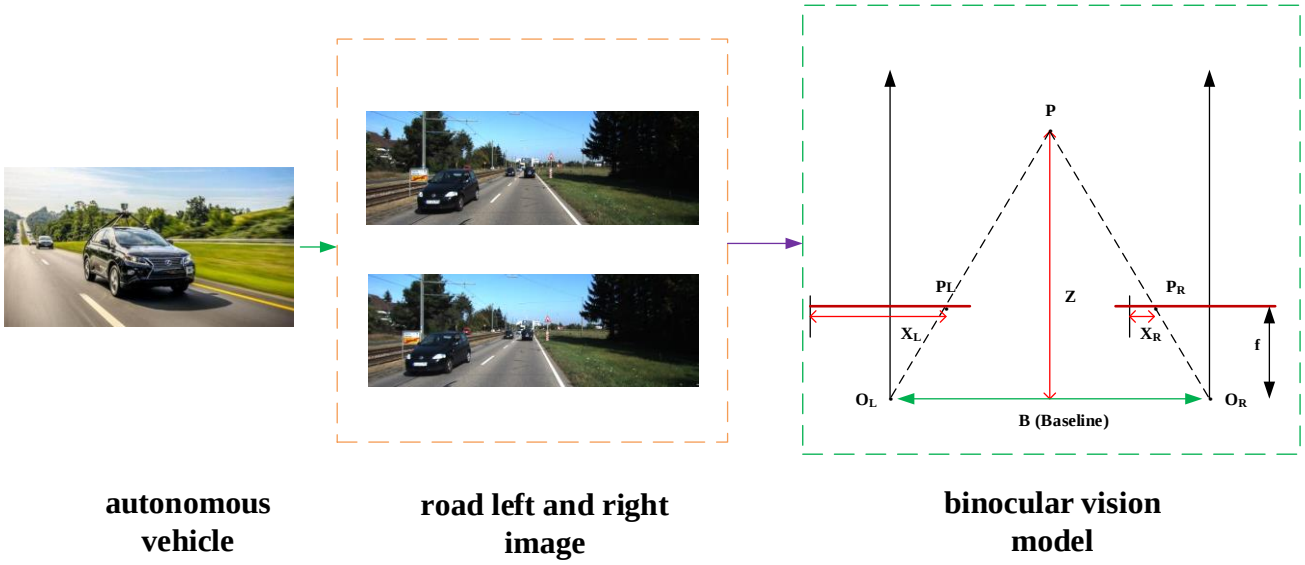



**Figure 1.** Workflow of binocular autonomous vehicle.

As shown in Figure 1, $f$ denotes the focal length of the binocular, indicating the distance from point $p$ to point $p_l$ and $p_r$. $d$ represents the difference between $p_l$ and $p_r$ in the x-axis direction. $B$ denotes the distance between two camera centers. $Z$ is the depth distance from point $p$ to point $O_l$ or $O_r$, which can be calculated by Eq (1.1):

$$Z = \frac{B * f}{d} \tag{1.1}$$

Stereo matching [28,29] is one of the most important and difficult parts of the stereo vision

algorithm, with a popular four-step pipeline being developed. This pipeline includes matching cost calculation, matching cost aggregation, disparity calculation, and disparity refinement. Traditional stereo matching methods can be roughly divided into three categories: global matching (GM) [30–32], local matching [33,34] and semi-global matching (SGM) [35–37]. The global matching method attempts to solve the optimal solution in the global, but the calculation is very laborious and the global optimal solution may not be found. The global method does not require the cost aggregation step. In addition, the selection of different calculation methods and optimal strategies have a great influence on the global method. The local matching algorithm matches local features within a certain range of matching points, which is depend on the rationality of the matching window. And local matching algorithm performance worse when processing the weak texture area and the occlusion area. Semi-global block matching (SGBM) is a combination of the global method and local method. SGBM adopts a pixel-by-pixel matching cost calculation method and a dynamic programming algorithm to achieve optimal path search in the one-dimensional smoothing constraints. However, the performance of the traditional stereo matching methods is severely limited by the handcrafted features adopted by cost functions, which cannot meet the demands of computation accuracy in complex scenes.

With the development of artificial intelligence, increasing researchers have begun to attempt to solve the stereo matching problem using deep learning methods. LeCun [38] trained a convolutional neural network to compute the stereo matching cost. The matching cost is refined by cross-based cost aggregation and semi-global matching. Subsequently, Zbontar [39] presented Siamese convolutional neural network architecture for learning a similarity measure on image patches and applied them to the problem of stereo matching. Siamese network architecture has two effects: One tuned for speed, the other for accuracy. The output of the convolutional neural network is used to initialize the stereo matching cost. However, the above network structures are too complex and have application limitations in the autonomous driving scenarios.

Aiming at improving the matching accuracy of the network training for autonomous driving scenarios, an end-to-end stereo matching algorithm based on improved convolutional neural network is proposed for autonomous driving applications. The main contributions of our work are summarized as follows:

- An end-to-end training algorithm pipeline based on improved convolutional neural network for autonomous driving application is designed.
- The “downsize” full-connection layer combined with subsequent network optimize is employed in the proposed network to improve the efficiency of the algorithm.
- A batch normalization layer is introduced after each convolutional layer to accelerate the convergence of the proposed network for autonomous scenario training with a large learning rate.
- An improved loss function is adopted to approximate the similarity of positive and negative samples in a more relaxed constraint and improve the matching effect of the outputs.

The KITTI 2012 and KITTI 2015 datasets are used to verify the effeteness and robustness of the proposed algorithm. Experimental results show that the end-to-end algorithm has a better performance in stereo matching to obtain a dense disparity map for the binocular vision system in autonomous driving.

## 2. Related works

End-to-end deep stereo networks have not been extensively studied until the first large scale synthetic stereo datasets disclosed by Mayer et al. [40]. There are lots of traditional stereo matching methods were proposed in recent years, such as GM, SGM, SBGM and so on. Specifically, SGM is a method based on block similarity while SBGM is a method based on disparity learning. Different from the traditional matching cost calculation method, Zbontar [39] and Lecun [38] firstly employed convolutional neural network named it MC-CNN to learn the similarity measurement between two image patches and used it to initialize the matching cost. Luo et al. [41] proposed a Siamese network which can produce excellent accurate results in less than a second of GPU computation. However, our method does not focus on low-level vision tasks such as optical flow, so the above methods are not appropriate. Shaked and Wolf [42] deepened the network for matching cost calculation using a highway network architecture with multi-level weighted residual shortcuts. It was demonstrated that this architecture outperformed several networks, such as the base network from MC-CNN.

Subsequently, Kendall et al. [43] proposed a method for learning parallax based on three dimensional (3D) convolution end-to-end. The method uses the geometrical characteristics of the image to extract depth features, the 3D convolution layer in the network is used to improve disparity estimation to realize parallax learning of sub-pixel precision without post-processing. However, the road matching effect in the above method is not satisfactory in complex scenes. Guney et al. [44] designed the Displet network, which uses sparse parallax estimation and image semantic segmentation to normalize the parallax, which can effectively solve the mismatching problem of stereo matching. Mayer et al. [40]. presented a novel approach named DispNet, which is an end-to-end CNN using synthetic stereo pairs for training. In parallel with the proposal of DispNet, similar CNN architectures are also applied to optical flow estimation, leading to FlowNet [45] and its successor (FlowNet 2.0 [46]). GWCNet [47] introduces group-wise correlation to provide better similarity measures than previous work and it can cooperate with the concatenation volume to further improve the performance.

After that，Pang et al. [48] proposed a cascaded residual learning (CRL) network with richer input information, which consists of two parts: DisFullNet and DisResNet. Specifically, the input of DisResNet includes both the output disparity maps of the DisFullNet network and the left graph generated by the warp operation, such a network structure can not only improve the training efficiency, but also optimize the initial parallax map, however, the matching accuracy is lower than FlowNet. Different form CRL, Liang et al. [49]. propose to calculate reconstruction error in feature space rather than color space and share features between disparity estimation network and refinement network.

To improve the matching accuracy of the network training for autonomous driving scenarios, an end-to-end stereo matching algorithm based on an improved convolutional neural network is proposed, which has a better performance in stereo matching. In addition, a better dense disparity map can be obtained in the proposed method for binocular vision system in autonomous driving.

## 3. The pipeline of the proposed algorithm

The proposed algorithm is used to extract the depth information of autonomous driving scenarios utilizing the images captured by the left and right cameras carried in cars. The flow chart is shown in Figure 2. in which L and R represent the left images and the right images captured by the binocular camera respectively. Firstly, taking the image pairs and corresponding labels as inputs of

the proposed neural network in both training and testing stages. Then the relevant parameters are set and the training model is obtained after several iterations. Finally, the trained feature model is loaded to obtain the depth information, which is represented by a dense disparity map.

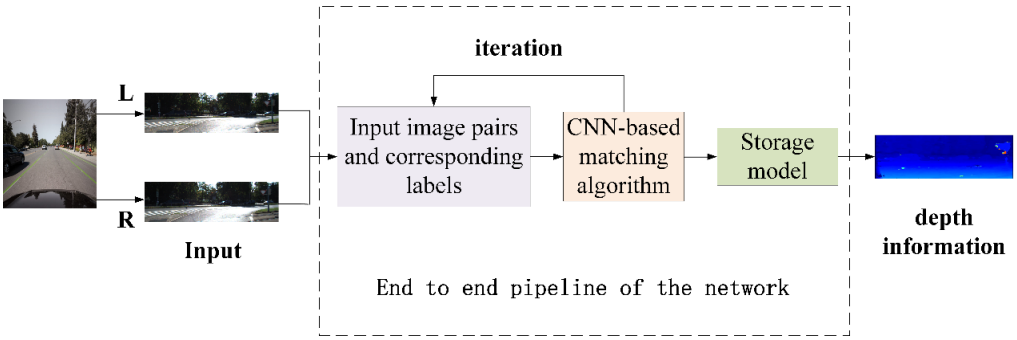



**Figure 2.** The flow chart of the proposed algorithm.

### *3.1. Computing the matching cost*

Generally, the first step of the stereo matching method is to compute the matching cost at each position for each disparity image. The purpose of this process is to find the corresponding pixel points in matching maps.

We use deep learning methods to compute the matching cost in the stereo matching procedure instead of finding matching points between the left and right pairs. The images are divided into multiple blocks, the red box (a) and (b) in Figure 3 are the blocks in the left and right images respectively, in which $P$ and $Q$ denote one pixel in the corresponding block separately. By comparing the similarities of the blocks to find the corresponding pixel points, the similarity can be calculated by the sum of absolute differences (SAD) in the image blocks. As shown in Eq (3.1), the smaller the similarity, the larger the matching cost.

$$D(x,y)=\sum_{s=1}^{M}\sum_{t=1}^{N}\left|P(x+s-1,y+t-1)-Q(s,t)\right| \tag{3.1}$$

(a) (b)

**Figure 3.** Matching based on patches. (a) patch in the left image; (b) patch in the right image.

### *3.2. Network architecture*

The architecture of the proposed network is depicted in Figure 4. The convolutional neural network layer is used to extract different feature information of the left and right image blocks. A batch normalization layer is added after each convolutional layer to speed up the convergence of network training. Meanwhile, batch normalization is employed to keep the training process more smoothly with a higher learning rate and less initialization. Relu function [50,51] with a stronger expression ability is adopted to maintain the convergence speed of the model in a stable state. After four iterations, the left and right features are concatenated into the full connection layer. There are 512, 384, 256, and 128 neurons respectively in the "downsize" full connection layer, which can be calculated by Eq (3.2):

$$n_i = \begin{cases} 512, & i = 1 \\ n_{i+1} + 128, & i = 2,3,4 \end{cases} \tag{3.2}$$

Where $n_i$ represents the number of neurons, $i$ denotes the number of iterations. In our experiment, the algorithm has the best matching effect when the number of iterations is 4.

The feature is classified and judged by a 4-layer full connection layer. A cross entropy cost function is employed in the training process, as shown in Eq (3.3):

$$F = t\log(s) + (1-t)\log(1-s) \tag{3.3}$$

Where $s$ is the output of the similarity comparison network, $t$ denotes the sample tag, $t = 1$ and $t = 0$ represent a positive input sample and a negative input sample separately.

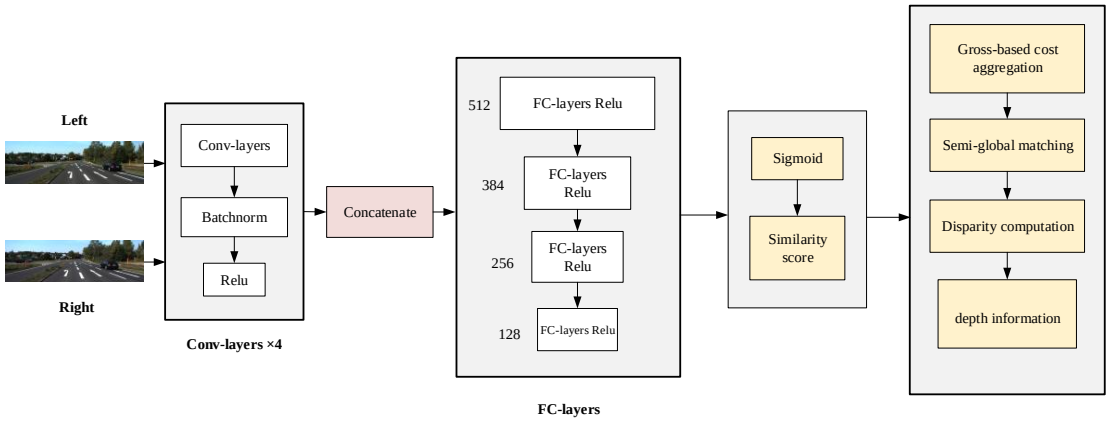



**Figure 4.** Structure of the network.

Similar/dissimilar judgment results are output in the form of scores for subsequent stereo matching. The cost function is constructed based on the inverse proportion of the similarity score, the more similar the image block, the smaller the cost. Finally, the cost function is post-processed by cross-cost aggregation and semi-global matching to choose the minimum cost as parallax. Since the

accuracy of the previous results could not meet the requirements of the experiment, the parallax was recalculated by fitting the conic through the adjacent cost at the adjacent parallax and calculate the final disparity maps.

### *3.3. Loss function design*

The KITTI 2012 and KITTI 2015 stereo matching datasets are chosen to construct a binary classification dataset. At each image pixel point where the real parallax is known, a positive sample and a negative sample are extracted respectively to ensure the dataset contains the same number of positive and negative samples. In order to enrich the data of the positive and negative samples, a positive sample $Q_{pos}$ is obtained by setting the intermediate pixel points of the right image block according to Eq (3.4):

$$Q_{pos} = (x - d + o_{pos}, y) \tag{3.4}$$

Where $o_{pos}$ is a random number in the range $[-v_{pos}, v_{pos}]$, which is generated during each iteration randomly. According to the experiments, it is confirmed that the matching effect is better when $v_{pos}$ is set to 4. Similarly, the negative sample $Q_{neg}$ is designed according to Eq (3.5):

$$Q_{neg} = (x - d + o_{neg}, y) \tag{3.5}$$

Where $o_{neg}$ is a random number in the range of $[-z_{low}, z_{high}]$, based on the generated $o_{pos}$, $z_{low}$ and $z_{high}$ are set to 4 and 18 respectively, which relates to the stereo matching algorithm that was used later. When the matching cost of the correct match and the approximate correct match is small, the cross-cost aggregation performs better. In the subsequent stage of network training, the benchmark disparity for the positive and negative sample classification design is $[1, 0]$, for each positive sample, the network expects it approach the similarity 1, and the similarity of negative samples is getting closer to 0.

In this paper, the positive and negative samples in the algorithm are presented in order to satisfy the perfect matching of image blocks. When the network is about to convergence, the cross-entropy loss function is utilized to fit the network, which can be described by the following cross entropy loss function:

$$D_{loss} = \begin{cases} t\ln(\Delta + s) + (1-t)\ln(1-s), t = 1 1 - 3\Delta \le s \le 1 - \Delta \\ t\ln(s) + (1-t)\ln(\Delta + 1 - s), t = 0, \Delta \le s \le 3\Delta \\ t\ln(s) + (1-t)\ln(1-s), others \end{cases} \tag{3.6}$$

As shown in Eq (3.6), $s$ represents the output of the network; $t = 1$ denotes positive samples and $t = 0$ is negative samples. In our experiment, $\Delta$ is set to 0.05. The algorithm performs better when the positive sample similarity is close to 1 and the negative sample similarity is close to 0.

## 4. Experimental results

### *4.1. Training KITTI datasets*

Small-batch gradient descent is adopted during the training process, after a certain number of training-validation iterations, the batch size is set to 150 and the momentum is set to 0.9. In neural network training, the learning rate is one of the most important factors affecting training speed and training accuracy. If the learning rate is too small, convergence is easy to guarantee, but the convergence speed will be slower; if the learning rate is too large, the learning speed is fast, but it may cause gradient disappearance [52] or gradient explosion [53]. The learning rate in our experiments is set to 0.02 in the training process. In order to fit the correction range of weights in different stages, the learning rate is reduced gradually in the later iteration. When the 18th epoch is completed (1 epoch is to train all samples through the network once), the loss function is close to convergence. The experimental comparison found that the test results are the best when the block size is set to $9 \times 9$, so the subsequent experiments are based on $9 \times 9$ blocks. The experimental platform is NVIDIA K80 based on the TensorFlow environment.

### *4.2. Experiment analysis*

#### 4.2.1. Loss function error comparison

From left to right in Table 1, the data are the error comparisons of training KITTI 2012 to validate KITTI 2012, training KITTI 2015 to validate KITTI 2015, training KITTI 2012 to validate KITTI 2015, and training KITTI 2015 to validate KITTI 2012, respectively. Table 1 shows the comparison of the improved loss function and MC-CNN [39] loss function based on KITTI 2012 and KITTI 2015 datasets.

**Table 1.** Loss function error comparison.

| Method | KITTI 2012 | KITTI2015 | KITTI 2012 on KITTI 2015 | KITTI 2015 on KITTI 2012 |
|---|---|---|---|---|
| MC-CNN | 2.63% | 3.27% | 4.03% | 3.93% |
| Ours | 2.62% | 3.26% | 4.01% | 3.88% |

As shown in Table 1, our method achieves the lowest error in the four experiments, which proves the validity of the proposed loss function. Table 2 gives the final parallax error comparison obtained by different $\Delta$ values in the improved loss function, which can be seen that the error result is the smallest when $\Delta$ is 0.05.

**Table 2.** Error comparison between different $\Delta$ values.

| $\Delta$ | Error |
|---|---|
| 0.03 | 3.262 |
| 0.04 | 3.267 |
| 0.05 | 3.251 |
| 0.06 | 3.283 |
| 0.07 | 3.268 |

#### 4.2.2. Comparison with other methods

The proposed algorithm is compared with traditional stereo matching algorithms using in autonomous driving scenarios, including Efficient Large-Scale Stereo Matching [54] (Elas), SGM [55], Slanted Planar Smoothing (SPS) [56], Fast R-CNN [57] Matching, and MC-CNN [39] Fast and Slow, the objective error indicators are used for the comparison of experimental results. As shown in Table 3, the indicator is the ratio of the pixel points, in which the disparity value to the reference disparity is greater than m (m = 2, 3, 4, 5) pixels. It can be seen from Tables 3 and 4 that the error value of the proposed algorithm is smaller than other algorithms when the pixel radio is large than 3 pixels, which indicates the effectiveness of the proposed algorithm.

In addition, Tables 3 and 4 also show qualitative results on the runtime of the proposed method in KITTI 2012 and KITTI 2015 separately, which can be observed that our method has a clear advantage in runtime compared with other algorithms.

**Table 3.** Error comparison of disparity with different algorithms (KITTI 2012).

| Algorithm | > 2 pixels | > 3 pixels | > 4 pixels | > 5 pixels | Runtime (s) |
|---|---|---|---|---|---|
| Elas | 22.71 | 21.08 | 20.24 | 19.56 | 43 |
| SGM | 6.27 | 4.88 | 4.15 | 3.56 | 48 |
| SPS | 4.86 | 3.78 | 3.16 | 2.78 | 21 |
| Fast R-CNN | 4.97 | 3.06 | 2.38 | 2.04 | 9 |
| MC-CNN-fast | 4.89 | 3.03 | 2.30 | 1.93 | 1.37 |
| MC-CNN-slow | 4.28 | 2.62 | 2.03 | 1.73 | 1.55 |
| Proposed | 4.37 | 2.60 | 2.01 | 1.71 | 0.98 |

**Table 4.** Error comparison of disparity with different algorithms (KITTI 2015).

| Algorithm | > 2 pixels | > 3 pixels | > 4 pixels | > 5 pixels | Runtime (s) |
|---|---|---|---|---|---|
| Elas | 24.08 | 19.22 | 17.58 | 16.83 | 40 |
| SGM | 10.04 | 6.94 | 5.46 | 4.49 | 49 |
| SPS | 7.15 | 4.57 | 3.47 | 2.94 | 18 |
| Fast R-CNN | 6.79 | 4.37 | 2.57 | 2.04 | 13 |
| MC-CNN-fast | 7.54 | 4.01 | 2.84 | 2.32 | 1.88 |
| MC-CNN-slow | 6.38 | 3.28 | 2.37 | 1.97 | 3 |
| Proposed | 6.56 | 3.26 | 2.34 | 1.93 | 1.02 |

#### 4.2.3. Performance of the proposed algorithm

In Figures 5–8, we show qualitative parallax results of our method on KITTI 2012 and KITTI 2015. By learning our method is often able to handle challenging scenarios, such as exposure and backlit autopilot scenes. The error map refers to the difference between the calculated disparity map and the pixel points of the reference disparity. It can be seen from the disparity map that the algorithm can obtain a smooth and dense disparity map, especially in the edge region of the target, and the edge information of the original target is retained obviously.

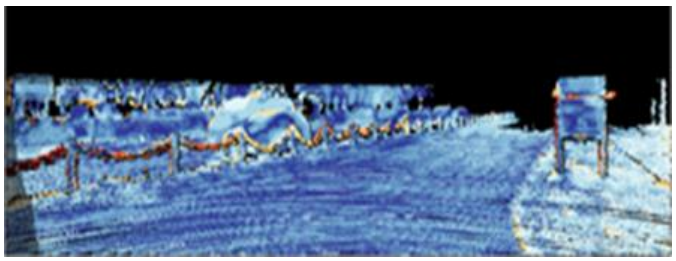
In addition, our method also has effect on exposure and backlit scenes. For example, in Figure 5a,b, although the car in the shade, our method can still get a better dense parallax map and error map, as

shown in Figure 5c,d. For the exposure road in Figure 6a,b, our method can obtain the clear dense parallax map and error map, as shown in Figure 6c,d.

(a)

(b)

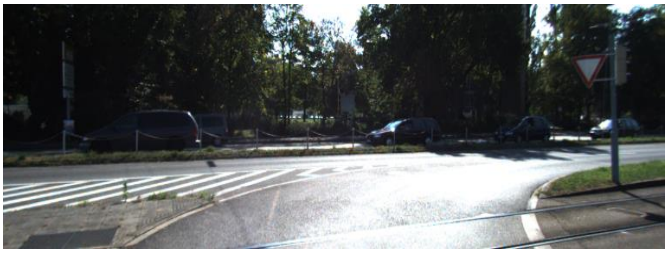
(c)



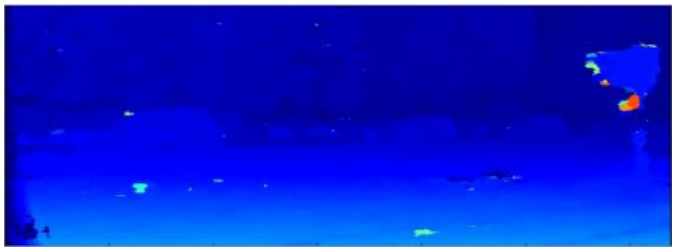
(d)

**Figure 5.** Disparity map extracts from right and left image (a) left original image (b) right original image (c) disparity prediction (d) error map.

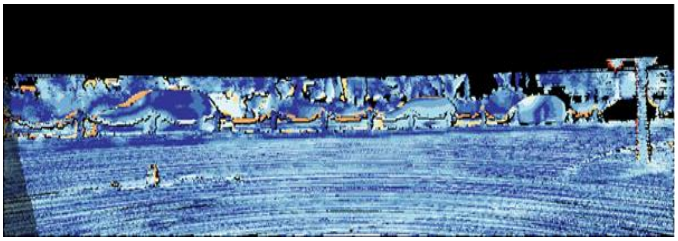
(a)



(b)



(c)

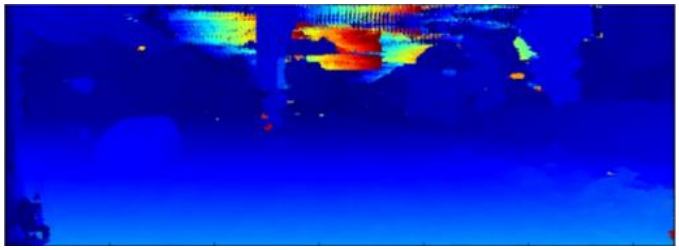


(d)

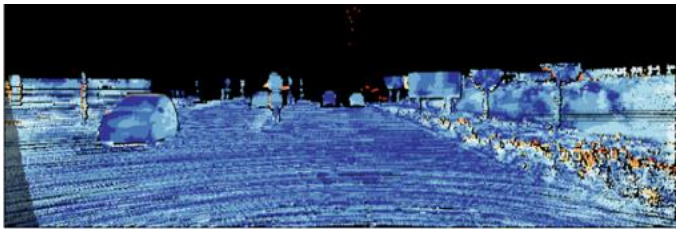
**Figure 6.** Disparity map extracts from right and left image (a) left original image (b) right original (c) disparity prediction (d) error map.

(a)

(b)



(c)

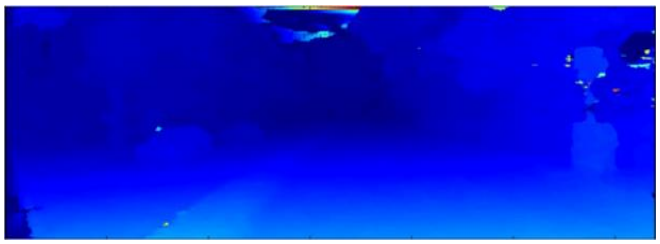


(d)

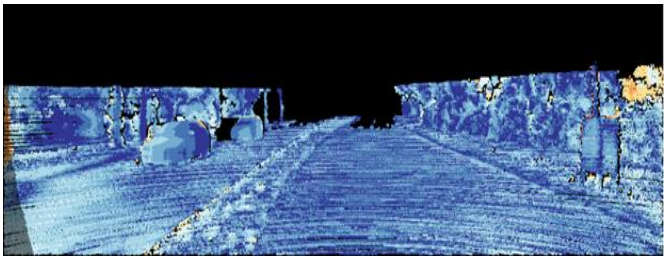
**Figure 7.** Disparity map extracts from right and left image (a)left original image(b)right original image (c)disparity prediction (d)error map.

(a)

(b)



(c)



(d)

**Figure 8.** Disparity map extracts from right and left image (a)left original image(b)right original image (c)disparity prediction (d)error map.

## 5. Conclusions

In the field of autonomous driving, the accurate computation of depth information is crucial to

the driving safety. Recent studies using CNNs for stereo matching have achieved prominent performance. In this paper, an end-to-end stereo matching algorithm based on the improved convolutional neural network structure is proposed for autonomous driving scenarios. A "downsize" full connection layer and an improved loss function are introduced in the proposed network. The experiments are carried out on the KITTI 2012 and KITTI 2015 datasets. The experimental results show that our algorithm has higher accuracy and efficiency compared with the traditional algorithms and some deep learning-based methods. The loss function error of the proposed method for KITTI 2012 and KITTI 2015 datasets are reduced to 2.62 and 3.26% respectively, which also reduces the runtime of the proposed algorithm. The end-to-end stereo matching method proposed in this paper may provide the location algorithm supports for the automatic driving area. For future work, we plan to focus on using segmentation information to optimize parallax images after matching for more complicated driving tasks.

## Acknowledgments

This research was funded by the National Natural Science Foundation of China, grant numbers 61605175, 61602423 and the Department of Science and Technology of Henan Province, China, grant numbers 192102210292, 182102110399.

## Conflict of interest

All authors declare no conflicts of interest in this paper.

## References

1. S. He, Z. Li, Y. Tang, Z. Liao, F. Li, S. Lim, Parameters compressing in deep learning, *Comput. Mater. Continua*, **62** (2020), 321–336.
2. D. Zeng, Y. Dai, F. Li, J. Wang, A. K. Sangaiah, Aspect based sentiment analysis by a linguistically regularized CNN with gated mechanism, *J. Intell. Fuzzy Syst.*, **36** (2019), 1–10.
3. R. Meng, S. G. Rice, J. Wang, X. Sun, A fusion steganographic algorithm based on faster R-CNN, *Comput. Mater. Continua*, **55** (2018), 1–16.
4. S. Zhou, M. Ke, P. Luo, Multi-camera transfer GAN for person re-identification, *J. Visual Commun. Image Repres.*, **59** (2019), 393–400.
5. Y. Song, G. Yang, H. Xie, D. Zhang, X. Sun, Residual domain dictionary learning for compressed sensing video recovery, *Multimedia Tools Appl.*, **76** (2017), 10083–10096.
6. W. Huang, Y. Xu, X. Hu, Compressive hyperspectral image reconstruction based on spatial-spectral residual dense network, *IEEE Geoence Remote Sens. Lett.*, **17** (2020), 884–888.
7. J. Zhang, X. Jin, J. Sun, J. Wang, A. K. Sangaiah, Spatial and semantic convolutional features for robust visual object tracking, *Multimedia Tools Appl.*, **79** (2020), 15095–15115.
8. Z. Lu, B. Xu, L. Sun, T. Zhan, S. Tang, 3D channel and spatial attention based multi-scale spatial spectral residual network for hyperspectral image classification, *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, **13** (2020), 1–1.
9. L. Sun, C. Ma, Y. Chen, Y. Zheng, H. J. Shim, Z. Wu, Low rank component induced spatial-spectral kernel method for hyperspectral image classification, *IEEE Trans. Circuits Syst. Video Technol.*, **30** (2019), 1–1.

10. W. Huang, Y. Huang, H. Wang, Local binary patterns and superpixel-based multiple kernels for hyperspectral image classification, *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, **13** (2020), 4550–4563.
11. N. Sünderhauf, O. Brock, W. Scheirer, R. Hadsell, D. Fox, J. Leitner, et al., The Limits and potentials of deep learning for robotics, *Int. J. Rob. Res.*, **37** (2018), 405–420.
12. G. Giordano, M. Segata, F. Blanchini, R. L. Cigno, O. Altintas, C. Casetti, et al., A joint network/control design for cooperative automatic driving, *IEEE Access*, 2017.
13. S. Belciug, F. Gorunescu, Error-correction learning for artificial neural networks using the bayesian paradigm. Application to automated medical diagnosis, *J. Biomed. Inf.*, **52** (2014), 329–337.
14. B. Sayed, I. Traoré, A. Abdelhalim, IF-Transpiler: inlining of hybrid flow-sensitive security monitor for javascript, *Comput. Secur.*, **75** (2018), S0167404818300397.
15. D. I. D. Han, M. C. T. Dieck, T. Jung, Augmented Reality Smart Glasses (ARSG) visitor adoption in cultural tourism, *Leisure Stud.*, **38** (2019), 1–16.
16. N. Sünderhauf, O. Brock, W. Scheirer, R. Hadsell, The limits and potentials of deep learning for robotics, *Int. J. Rob. Res.*, **37** (2018), 405–420.
17. L. Matthies, R. Brockers, Y. Kuwata, S. Weiss, Stereo vsion-based obstacle avoidance for micro air vehicles using disparity space, *IEEE Int. Conf. Rob. Automation*, (2014), 3242–3249
18. S. O. Escolano, C. Rhemann, S. R. Fanello, W. Chang, A. Kowdle, Y. Degtyarev, et al., Holoportation: virtual 3D teleportation in real-time, *User Interface Software Technol.* (2016), 741–754.
19. C. Guindel, D. Martín, J. M. Armingol, Traffic scene awareness for intelligent vehicles using ConvNets and stereo vision, *Rob. Auton. Syst.*, **112** (2019), 109–122.
20. Y. J. Lee, M. W. Park, 3D tracking of multiple onsite workers based on stereo vision, *Autom.Constr.*, **98** (2019), 146–159.
21. V. Gabelica, S. Livet, F. Rosu, Optimizing native ion mobility Q-TOF in helium and nitrogen for very fragile noncovalent interactions, *J. Am. Soc. Mass Spectrom.*, **29** (2018), 2189–2198.
22. N. Aslani, G. Janbabaei, M. Abastabar, J. F Meis, M. Babaeian, S. Khodavaisy, et al., Identification of uncommon oral yeasts from cancer patients by MALDI-TOF mass spectrometry, *Bmc Infect. Dis.*, **18** (2018), 24.
23. Z. Song, High-speed 3D shape measurement with structured light methods: A review, *Opt. Lasers Eng.*, **106** (2018), 119–131.
24. C. Jiang, B. Lim, S. Zhang, Three-dimensional shape measurement using a structured light system with dual projectors, *Appl. Opt.*, **57** (2018), 3983.
25. L. Gang, H. Song, L. Chan, Matching algorithm and parallax extraction based on binocular stereo vision, *Smart Innovations Commun. Comput. Sci.*, (2019), 347–355
26. A. L. Webber, J. M. Wood, B. Thompson, E. E Birch, From suppression to stereoacuity: a composite binocular function score for clinical research, *Ophthalmic Physiol. Op.*, **39** (2019), 53–62.
27. Q. Xie, Q. Long, L. Zhang, Z. Sun, A robust real-time computing-based environment sensing system for intelligent vehicle, *Comput. Vision Pattern Recognit.*, 2020.
28. G. Zhang, D. Zhu, W. Shi, X.Ye, J. Li, X.Zhang, et al., Multi-dimensional residual dense attention network for stereo matching, *IEEE Access*, **7** (2019), 1–1.
29. H. Y. Lai, Y. H. Tsai, W. C. Chiu, Bridging stereo matching and optical flow via spatiotemporal correspondence, *IEEE Conf. Comput. Vision Pattern Recognit.*, (2019), 1890–1899.

30. M. Ye, J. Li, A. J. Ma, L. Zheng, P. C. Yuen, Dynamic graph co-matching for unsupervised video-based person re-identification, *IEEE Trans. Image Process.*, **28** (2019), 1–1.
31. C. Le, L. Xin, Sparse3D: A new global model for matching sparse RGB-D dataset with small inter-frame overlap, *Comput.-Aided Des.*, **102** (2018), S0010448518302276.
32. A. Klaus, M. Sormann, K. Karner, Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure, *IEEE Int. Conf. Pattern Recognit.*, **3** (2006), 15–18.
33. D. Wang, H. Liu, X. Cheng, A miniature binocular endoscope with local feature matching and stereo matching for 3D measurement and 3D reconstruction, *Sensing*, **18** (2018), 2243.
34. C. L. Mills, R. Garg, J. S. Lee, Functional classification of protein structures by local structure matching in graph representation, *Protein Sci. Publ. Protein Soc.*, **27** (2018), 1125–1135.
35. Y. Anisimov, O. Wasenmüller, D. Stricker, Rapid light field depth estimation with semi-global matching, *Comput. Vision Pattern Recognit.*, 2019.
36. T. Y. Chuang, H. W. Ting, J. J. Jaw, Dense stereo matching with edge-constrained penalty tuning, *IEEE Geosci. Remote Sens. Lett.*, **15** (2018), 1–5.
37. A. Seki, M. Pollefeys, Sgm-nets: Ssemi-global matching with neural networks, *IEEE Conf. Comput. Vision Pattern Recognit.*, (2017), 231–240.
38. J. bontar, Y. Lecun, Computing the stereo matching cost with a convolutional neural network, *IEEE Conf. Comput. Vision Pattern Recognit.*, 2015.
39. J. Žbontar, Y. Lecun, Stereo matching by training a convolutional neural network to compare image patches, *Comput. Vision Pattern Recognit.*, 2015.
40. N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A.Dosovitskiy, et al., A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation, *IEEE Comput. Vision Pattern Recognit*., 2016.
41. Y. Feng, Z. Liang, H. Liu, Efficient deep learning for stereo matching with larger image patches, *Int. Congr. Image Signal Process. BioMed. Eng. Inf.*, 2017.
42. A. Shaked, L. Wolf, Improved stereo matching with constant highway networks and reflective confidence learning, *Comput. Vision Pattern Recognit.*, (2017), 6901–6910.
43. A. Kendall, H. Martirosyan, S. Dasgupta, P. Henry, R. Kennedy, A. Bachrach, et al., End-to-End learning of geometry and context for deep stereo regression, *Comput. Vision Pattern Recognit.*, 2017.
44. F. Guney, A. Geiger, Displets: Resolving stereo ambiguities using object knowledge, *IEEE Conf. Comput. Vision Pattern Recognit.*, 2015.
45. P. Fischer, A. Dosovitskiy, E. Ilg, P. Häusser, C. Hazırbaş, V. Golkov, et al., FlowNet: Learning optical flow with convolutional networks, *Deep Learn. Inverse Prob.*, 2015.
46. E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, T. Brox, FlowNet 2.0: Evolution of optical flow estimation with deep networks, *IEEE Conf. Comput. Vision Pattern Recognit.*, 2017.
47. X. Guo, K. Yang, W. Yang, X. Wang, H. Li , Group-wise correlation stereo network, *Comput. Vision Pattern Recognit.*, 2019.
48. J. Pang, W. Sun, J. S. Ren, C. Yang, Q. Yan, Cascade residual learning: A two-stage convolutional neural network for stereo matching, *Comput. Vision Pattern Recognit.*, 2017.
49. Z. Liang, Y. Guo, Y. Feng, W. Chen, L. Qiao, L. Zhou, et al., Stereo matching using multi-level cost volume and multi-scale feature constancy, *IEEE Trans. Pattern Anal. Mach. Intell.*, **99** (2019), 1–1.

50. J. Schmidt-Hieber, Nonparametric regression using deep neural networks with ReLU activation function, *Stat. Theory*, **48** (2017), 1875–1897.
51. S. Woo, C. L. Lee, Decision boundary formation of deep convolution networks with ReLU, *IEEE Comput. Vision Pattern Recognit.*, 2018.
52. E. Özarslan, C. Yolcu, M. Herberthson, H. Knutsson, C. Westin, Influence of the size and curvedness of neural projections on the orientationally averaged diffusion MR signal, *Front. Phys.*, **6** (2018), 17.
53. Z. Chen, L. Deng, B. Wang, G. Li, Y. Xie, A comprehensive and modularized statistical framework for gradient norm equality in deep neural networks, *IEEE Tran. Pattern Anal. Mach. Intell*., 2020.
54. A. Geiger, M. Roser, R. Urtasun, Efficient large-scale stereo matching, *Comput. Vision*, (2010), 25–38.
55. H. Hirschmuller, D. Scharstein, Evaluation of stereo matching costs on images with radiometric differences, *IEEE Trans. Pattern Anal. Mach. Intell.*, **31** (2009), 1582–1599.
56. K. Yamaguchi, D. McAllester, R. Urtasun, Efficient joint segmentation, occlusion labeling, stereo and flow estimation, *Comput. Vision – ECCV*, (2014), 756–771.
57. S. Ren, K. He, R. Girshick, J. Sun, Faster R-Cnn: Towards real-time object detection with region proposal networks, *IEEE Trans. Pattern Anal. Mach. Intell.*, **39** (2015), 91–99.



AIMS Press