Peer-reviewed version available at Appl. Sci. 2019, 9, 1065; doi:10.3390/app9061065

Article

A Novel Interactive Fusion Method with Images and Point Clouds for 3D Object Detection

Kai Xu [1,2], Zhile Yang [1], Yangjie Xu[1], Liangbing Feng [1,*]

[1] Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences Shenzhen 518055, Guangdong Province, China; {zl.yang&lb.feng}@siat.ac.cn

[2] School of Software Engineering University of Science and Technology of China 188 Renai Road, Suzhou 215123, Jiangsu Province, China; xk1992@mail.ustc.edu.cn

* Correspondence: lb.feng@siat.ac.cn

Abstract: This paper aims at tackling with the task of fusion feature from images and its corresponding point clouds for 3D object detection in autonomous driving scenarios basing on AVOD, an Aggregate View Object Detection network. The proposed fusion algorithms fuse features targeted from Bird's Eye View (BEV) LIDAR point clouds and its corresponding RGB images. Differs in existing fusion methods, which are simply the adoptions of concatenation module, element-wise sum module or element-wise mean module, our proposed fusion algorithms enhance the interaction between BEV feature maps and its corresponding images feature maps by designing a novel structure, where single level feature maps and another utilizes multilevel feature maps. Experiments show that our proposed fusion algorithm produces better results on 3D mAP and AHS with less speed loss comparing to existing fusion method used on the KITTI 3D object detection benchmark.

Keywords: fusion; point clouds; images; object detection

1. Introduction

It is the fact that deep neural networks rely on a large number of data to guarantee the training effectiveness [17]. In general, the more data is fed, the better performance will be obtained, particularly when feeding abundant sensor data to the network model. In field of self-driving cars or 3D object detection, camera and lidar are dominant sensors. RGB images from cameras contain rich texture information of the ambience, whereas the depth are lost. Point clouds from lidar can provide accurate depth and reflection intensity descriptions, but the resolution is comparatively low. Naturally, the effective fusion [19] of these sensors expects to deal with the drawbacks of single sensor in complicated driving scenarios.

There are three major fusion algorithms to solve multi-sensor fusion problem including early fusion [1][2][3][10][11]; late fusion [4][5][6] and deep fusion [7]. In the early fusion architecture, firstly, features from single sensor concatenate or element-wise sum (mean) the features from other sensors; secondly, the outputs of fused feature maps would send to classification or segmentation. An advantage of early fusion is that the joint feature space between the modalities is potentially more expressive. However, the learning problem becomes more difficult due to that the classifier must learn a mapping from a higher-dimensional feature space. Late fusion usually has multi-net-branch and each network branch is first run on its network structure in corresponding sensing modality separately, then feature maps from each branch would fuse by concatenation or element-wise sum (mean) as final input to classification or segmentation. Compared to the early fusion, late fusion is easier to learn, but less expressive and sometimes the former could utilize more data than the later one. Especially when training data is not sufficient, the late fusion performs more effective. The recent deep fusion inspired by [8][9] uses element-wise mean for the join operation. Besides, three branches, using fused feature as unify input, would be trained dependently then combined with element-wise mean and iteration. Deep fusion makes features from different views

interact frequently, but in each interaction, it is also linear model which is the same as early fusion and late fusion. Linear model is flexible and simple accomplishment, however, it is much less expressive than the nonlinear model, which may suffer from more time and memory cost.

Our proposal fusion algorithm aims at combining the linear model and the nonlinear model, and enhancing the interactions between image features and its corresponding point clouds features, and the independence of multi-view features is kept at the same time. The proposal fusion algorithm is elaborated in the following part.

2. Related Work and Proposed Method

There are a few works that exploit multi-sensor of data including the combination of RGB images and depth images and the fusion of RGB images and point clouds[17]. [11] utilizes RGB images and depth images with the early fusion strategy and trains pose-based classifiers for 2D detection[18]. [2][3] apply early fusion strategy by projecting point cloud to the plane and augmenting the image channels after upsampling. [6] fuses images and point clouds by late fusion strategy for urban segmentation and compute size, shape, position, color features, a high-dimension Bag-of-words (BoW) descriptor in images and point clouds. [7] fuse lidar bird view, lidar front view and image for 3D object detection with deep fusion. Besides, it project point cloud to the bird's eye view instead of image plane and get nice results in KITTI [16]; however, it perform poor in the test of average heading similarity (AHS) .

We evaluate our fusion algorithm in KITTI by 3D object detection with images and point clouds basing on AVOD [12]. Different from existing fusion algorithms, our fusion algorithms combine linear model and nonlinear model and enhance the interaction and independence. We can find detail visual structure for one of our proposal fusion algorithms in Figure 1. It could be observed that from two types or one group input feature maps to fusion output feature maps, our proposal fusion algorithms contain linear model element-wise mean, and concatenation and nonlinear model convolution with Relu [13] activation function. Besides, it reinforces the interaction and independence by adding one type feature maps into the other type and considering the mixing ratio, then two types of fused feature maps are through convolution layers separately. Finally, the two branches fuse again by element-wise mean. We name this framework as single level feature maps fusion algorithm. In addition, multilevel feature maps fusion algorithm is illustrated in Figure 2, where four type or two group input feature maps are adopted to fusion output feature maps. The proposed multilevel fusion algorithm contains linear model element-wise mean, as well as concatenation and nonlinear model convolution with Relu [13] activation function. Moreover, it reinforces interaction and independence by adding one type feature maps into the other type and consider the mixing ratio, then four type fused feature maps are input through convolution layers separately, finally, the four branches fuse again by element-wise mean.

We exam our proposed fusion algorithm by replacing the two fusion parts of AVOD [12] with our fusion algorithm. To be specific, single level feature maps fusion algorithm uses Figure 1 structure is to displace the second fusion part of AVOD [12], and multilevel feature maps fusion algorithm uses Figure 2 structure to displace the first fusion part of AVOD.

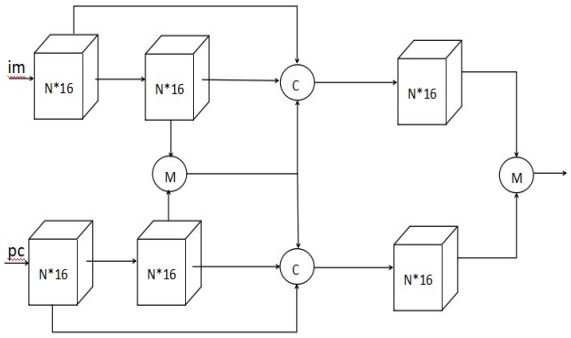

Figure 1. Visual structure of fusion algorithm with single level feature maps: firstly, images feature maps and point clouds feature maps fuse by element-wise mean. Then, the fused part concatenates with preceding two layer feature maps. Next, each of two branches is fed through convolution layer to reorganize feature maps, finally the two branches fuse again by element-wise mean to region proposal.

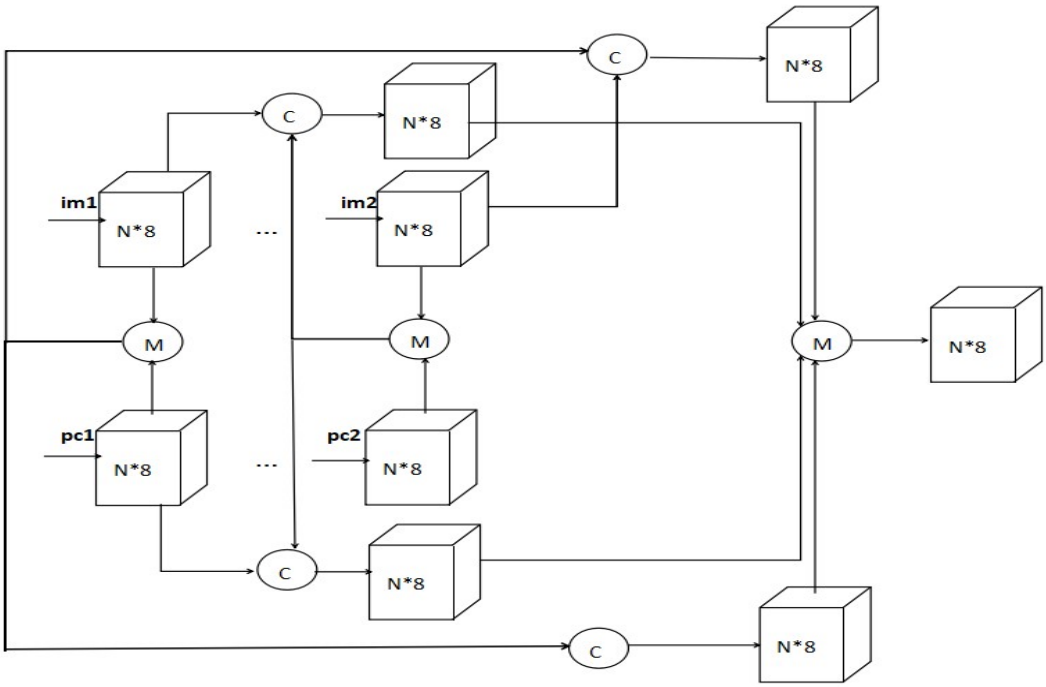

Figure 2. Visual structure of fusion algorithm with multilevel feature maps: firstly, images feature maps and point clouds feature maps in each group are fused by element-wise mean separately (im1 and pc1 are of one group, and im2 and pc2 are of another group). Then, the fused part concatenate with another corresponding level feature maps. Next, each of four branches is fed through convolution layer to reorganize feature maps. Finally, the four branches fuse again by element-wise mean to region proposal.

3. Experiments

We elevate our fusion algorithm on published AVOD [12] where the metrics are 3D AP (average precision) and AHS (average heading similarity). To test our proposed fusion algorithms,

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 12 February 2019 doi:10.20944/preprints201902.0105.v1

Peer-reviewed version available at Appl. Sci. 2019, 9, 1065; doi:10.3390/app9061065

we replace AVOD[12] data fusion parts by our proposed fusion algorithms and use default hyper-parameter, training set and validation set the same with AVOD to eliminate other influences and the performance on KITTI val set are shown in Table I, Table II , Figure 3 and Figure 4.

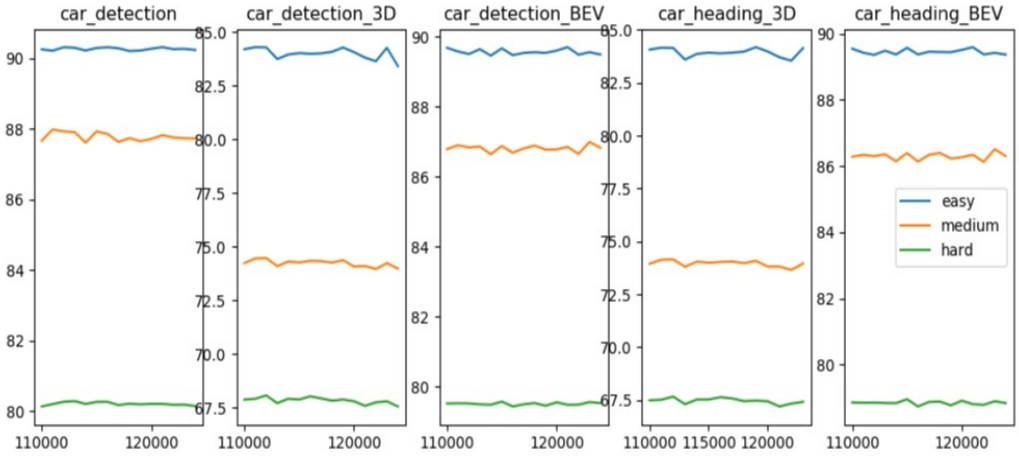

Figure 3. multilevel fusion algorithm AP vs. Step, and print the 5 highest performing checkpoints for each evaluation metric.

Figure 4_1

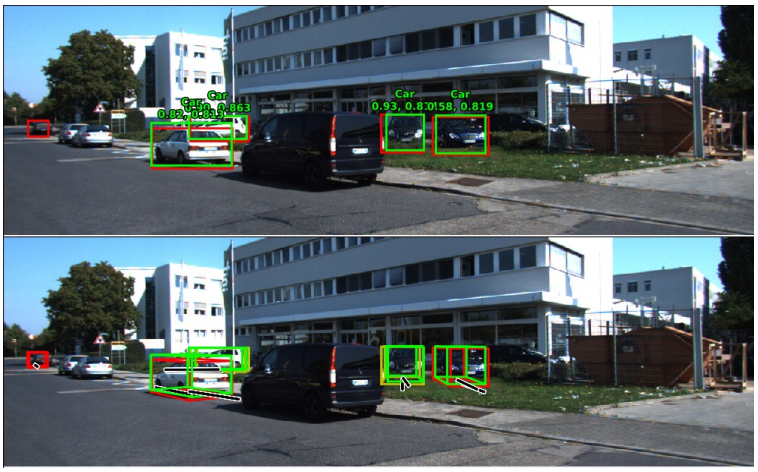

Figure 4_2

Figure 4_3

Figure 4_4

Figure 4_5

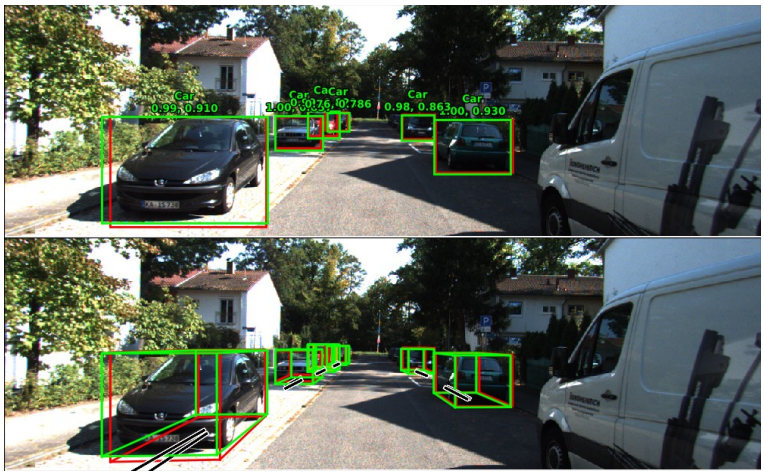

Figure 4_6

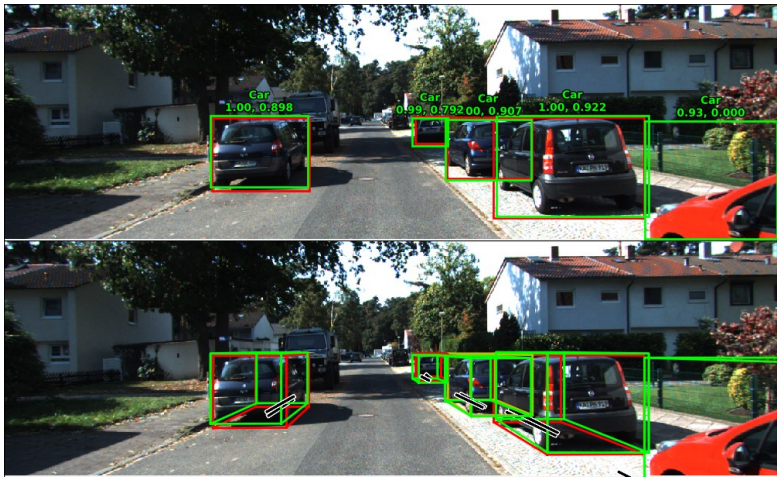

Figure 4_7

Figure 4. Visualization of multilevel fusion algorithm results on KITTI val set. Including 2D localization, category classification, 3D localization, orientation estimation, and category classification.

3.1. Kernel Design

Our proposed single level fusion algorithm, which is depicted by Figure 1, uses uniform convolution layers that are designed with kernel size 1*1, stride size 1, output feature map size is 16 and its input feature map comes from the last feature extraction layer whose feature map resolution is the same as original network input. Multilevel fusion algorithm, which is depicted by Figure 2, uses uniform convolution layers that are designed with kernel size 1*1, stride size 1, output feature map size 8 and its input feature maps come from the last feature extraction layer in which feature map resolution is the same as original network input and the last but one feature extraction layer of which feature map resolution is only the half of original network input.

In AVOD [12], feature map size cropped for proposal region is 3*3 and feature map size cropped for final prediction is 7*7. During training, we investigate how kernel with different size affects the performance and how does the kernel group inspiring by GoogLeNet [14] perform. The baseline 1 diverse kernel size multilevel fusion algorithm has 1*1, 3*3 kernel in first fusion part and 1*1, 3*3, 5*5 and 7*7 kernel in second fusion part. The baseline 2 diverse kernel size multilevel fusion algorithm has 1*1, 3*3 kernel in first fusion part and 1*1, 7*7 kernel in second fusion part. The results are show in Table 3. It could be seen that our approach has obtained the best in the ratio of feature maps size and zero padding. The higher ratio, which means that the number of zero by zero padding is equal or greater than the number of elements in feature maps, and the more disturbance to feature maps caused by zero is predominant so that the original distribution of feature maps is deviation to a large extent.

3.2. No BatchNorm

Batch normalization aims to eliminate the covariate shift in its input data which can improve the speed of learning and independence of each individual layer. However, we found that batch normalization regresses the 3D bounding box estimation performance. Therefore, our proposed fusion algorithms have no batch normalization layers.

3.3.Architecture Design Analysis

Our fusion algorithms aim to reinforces interaction and independence of different type of feature maps and utilize linear models and nonlinear models to enhance expression. The single level fusion algorithm can approximate a general function:

$$f_{im}^{[L+1]}\left(f_{im}^{[L-1]} \otimes f_{im}^{[L]} \otimes \left(f_{im}^{[L]} \otimes f_{pc}^{[L]}\right)\right) \otimes$$
$$f_{pc}^{[L+1]}\left(f_{pc}^{[L-1]} \otimes f_{pc}^{[L]} \otimes \left(f_{pc}^{[L]} \otimes f_{im}^{[L]}\right)\right)$$

where f[L+1]im is image feature maps of (L+1)th layer; f[L+1]pc denotes the point cloud feature maps of (L+1)th layer; $\otimes$ means concatenate or element-wise mean and the multilevel fusion algorithm can also approximate a general function:

$$f_1^{[L+1]}\left(\left(f_{im\,1}^{[L-K]} \otimes f_{pc\,1}^{[L-K]}\right) \otimes f_{im\,2}^{[L]}\right)$$
$$\otimes\ f_2^{[L+1]}\left(\left(f_{im\,1}^{[L-K]} \otimes f_{pc\,1}^{[L-K]}\right) \otimes f_{pc\,2}^{[L]}\right)$$
$$\otimes\ f_3^{[L+1]}\left(\left(f_{im\,2}^{[L]} \otimes f_{pc\,2}^{[L]}\right) \otimes f_{im\,1}^{[L-K]}\right)$$
$$\otimes\ f_4^{[L+1]}\left(\left(f_{im\,2}^{[L]} \otimes f_{pc\,2}^{[L]}\right) \otimes f_{pc\,1}^{[L-K]}\right)$$

where f[L+1] is the feature map of (L+1)th layer and subscript means different source of feature maps; $\otimes$ means concatenate or element-wise mean; K is an integer and less than L.

Table 1. Our proposed single level fusion algorithm evaluation on the car class in the validation set. For evaluation, we show the AP and AHS (in %) at 0.7 3D IoU .

	Easy		Moderate		Hard	
	AP	AHS	AP	AHS	AP	AHS
MV3D[15]	83.87	52.74	72.35	43.75	64.56	39.86
AVOD[12]	83.08	82.96	73.62	73.37	67.55	67.24
ours	**84.16**	**84.05**	**74.45**	**74.13**	**67.80**	**67.40**

Table 2. Our proposed multilevel fusion algorithm evaluation on the car class in the validation set. For evaluation, we show the AP and AHS (in %) at 0.7 3D IoU .

	Easy		Moderate		Hard	
	AP	AHS	AP	AHS	AP	AHS
MV3D[15]	83.87	52.74	72.35	43.75	64.56	39.86
AVOD[12]	83.08	82.96	73.62	73.37	67.55	67.24
ours	**84.62**	**84.41**	**74.88**	**74.45**	**68.30**	**67.79**

Table 3. Our proposed multilevel fusion algorithm evaluated on the car class in the validation set compared with baseline.

	Easy		Moderate		Hard	
	AP	AHS	AP	AHS	AP	AHS
baseline	83.02	82.84	73.71	73.13	67.79	67.15
baseline	84.02	83.84	74.42	74.03	68.16	67.74
ours	**84.62**	**84.41**	**74.88**	**74.45**	**68.30**	**67.79**

4. Conclusions

In this paper, we propose two fusion algorithms. One is single level feature maps fusion algorithm, and the other is multilevel feature maps fusion algorithm. Both of the two fusion algorithms do enhance interaction and independence between BEV feature maps and its corresponding images feature maps by designing a novel structure differentiated from existing fusion methods. Our proposed fusion algorithms define a nonlinear framework to improve potential expression. The nonlinear frameworks also take advantage of linear models, being similar to the existing fusion method, to flexible interaction and reducing cost of learning. Besides, the nonlinear frameworks can be easily embedded CNN network to make it utilized frequently. Experiments on the KITTI dataset show the effectiveness of our nonlinear fusion algorithms compared with existing fusion.

Acknowledgments: This work was supported in part by the Shenzhen Science and Technology Program under Grant No. JCYJ20170811160212033, and China NSFC under grants 61433012, U1435215.

References

1. Z. Cai, Q. Fan, R. Feris, and N. Vasconcelos. A unified multi-scale deep convolutional neural network for fast object detection. In ECCV, 2016.
2. T. Kim and J. Ghosh. Robust detection of non-motorized road users using deep learning on optical and lidar data. In Intelligent Transportation Systems (ITSC), 2016 IEEE,19th International Conference on, pages 271–276. IEEE, 2016.

3. S. Lange, F. Ulbrich, and D. Goehring. Online vehicle detection using deep neural networks and lidar based preselected image patches. In Intelligent Vehicles Symposium (IV),2016 IEEE, pages 954–959. IEEE, 2016.
4. J. Hoffman, S. Gupta, and T. Darrell. Learning with side information through modality hallucination. In CVPR, 2016.
5. S. Song and J. Xiao. Deep sliding shapes for amodal 3d object detection in rgb-d images. In CVPR, 2016.
6. R. Zhang, S. A.Candra, K. Vetter, Sensor Fusion for Semantic Segmentation of Urban Scenes, IEEE International Conference on Robotics & Automation - 2015.
7. X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, Multi-view 3d object detection network for autonomous driving, In IEEE CVPR, 2017.
8. G. Larsson, M. Maire, and G. Shakhnarovich. Fractalnet: Ultra-deep neural networks without residuals. arXiv:1605.07648, 2016.
9. J. Wang, Z. Wei, T. Zhang, and W. Zeng. Deeply-fused nets. arXiv:1605.07716, 2016.
10. C. Cadena and J. Kosecká. Semantic segmentation with heterogeneous sensor coverages, in ICRA, 2014.
11. M. Enzweiler and D. M. Gavrila. A multilevel mixture-of-experts framework for pedestrian classification. IEEE Transactions on Image Processing, 20(10):2967–2979, 2011.
12. J. Ku, M. Mozifian, J. Lee, A. Harakeh, S. Waslander. Joint 3D Proposal Generation and Object Detection from View Aggregation, arXiv: 1712.02294v3,2017.
13. X. Glorot, A. Bordes, and Y. Bengio. Deep sparse rectifier neural networks. In AISTATS, 2011.
14. W. Liu, Y. Jia, P. Sermanet, Scott Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich. Going Deeper with Convolutions. arXiv: 1409.4842v1.
15. X. Chen, H. Ma, J. Wan, B. Li, and T. Xia. Multi-view 3d object detection network for autonomous driving. In IEEE
CVPR, 2017.
16. Kitti 3d object detection benchmark leader board.
http://www.cvlibs.net/datasets/kitti/eval_object.php?obj_benchmark=3d. Accessed:2017-11-14 12PM.
17. Macher H, Landes T, Grussenmeyer P. From Point Clouds to Building Information Models: 3D Semi-Automatic Reconstruction of Indoors of Existing Buildings[J]. Applied Sciences, 2017, 7(10): 1030.
18. Tang C, Ling Y, Yang X, et al. Multi-View Object Detection Based on Deep Learning[J]. Applied Sciences, 2018, 8(9): 1423.
19. Yang J, Li S, Gao Z, et al. Real-Time Recognition Method for 0.8 cm Darning Needles and KR22 Bearings Based on Convolution Neural Networks and Data Increase[J]. Applied Sciences, 2018, 8(10): 1857.