Classification of overpass structures based on convolutional neural networks

Zhaoyang Xie

Shanghai Foreign Language School affiliated to SISU, Shanghai, China, 200080

zhaoyang_xie@126.com

Abstract. The navigation of overpass structures heavily relies on high-definition road maps, but in cases where these maps are unavailable, the automated classification system can enable the vehicle to identify potential road designs when access to satellite images is available, and navigate around the structure. In the paper, different Convolutional Neural Networks (LeNet5, AlexNet, and ResNet) are used for classification, and their effectiveness are compared. Data in the research is collected from Google Maps and Amap. The experiment results have shown that AlexNet has the best results, reaching 98% accuracy. ResNet is the second, reaching 96%. LeNet5 has the least accuracy, 75%. The classification model can easily identify the structure of common overpasses in cities as well as the countryside.

Keywords: road networks, overpass, classification, deep learning, classification, CNN.

1. Introduction

Road network is the most important part of a city, and road pattern recognition is very important in urban planning and automatic driving. Thus, in GIS research, the identification of road patterns becomes an important focus [1]. As overpass structures involve complex nodes, they are hard to navigate, and in some places, the road data is unmarked. Common overpass intersections include: Cloverleaf, Stack, Cloverstack, Turbine, and Trumpet. To increase the accuracy of classifying different overpass structures and compare the abilities of different CNN models(LeNet5, AlexNet, and ResNet), the paper uses satellite images from six cities and various highways from both Amap and Google Maps [2-3]. This paper utilizes different models of CNN to classify common overpass structures. The input only includes satellite image data and does not include any artificial labels. Because the road structure of the same class of overpass is relatively stable, the model can be used in automated driving, which enables the vehicle to predict the road structures ahead without the need for accurate online roadmaps.

2. Previous research

Previous identification of structures highly depend on road attributes. Scheiders used a method of combining road attributes and map data to identify structures [4]. In Xu's study, a relationship graph is used to identify the structural patterns [5]. Hao Li and Maosheng Hu used Faster-RCNN to identify different overpass structures in maps, but didn't classify them into groups [6]. Qian and He used CNN to classify road networks. Their work focuses on using AlexNet to identify the overpass structures of

DOI: 10.54254/2755-2721/6/20230523

trumpet, clover and cross. Their model doesn't need conventional road vector data, and reaches a high accuracy rate(98%) when identifying trumpet overpass structures [7]. However, the other common interchange structures(Cloverstack, Turbine, etc.) are included as 'other' data. Data are not enhanced and preprocessed for better robustness, and other models of CNN are not used for comparison.

3. Research method

3.1. Data and data preprocessing

In the paper, the 'Raw' data for training and testing are captured from Amap and Google Maps [2-3]. Common overpass intersections include: Cloverleaf, Stack, Cloverstack, Turbine and Trumpet. Figure 1 to 5 shows the examples of the above structures.

Figure 1. Cloverleaf Structure[3].
(G93 & Shuangnan Avenue Intersection, Chengdu, 30°36'38.1"N 103°56'52.8"E)

Figure 2. Stack Structure [3].
(Yan'an Elevated Road & North-South Elevated Road Intersection, Shanghai, 31°13'34.4"N 121°27'52.2"E)

Figure 3. Cloverstack Structure [2].
(G1501 & S1 Intersection, Shanghai, 31°11'20.3"N 121°45'14.8"E)

Figure 4. Cloverleaf(L) and Turbine(R) Structures [3].
(3rd Ring Road, Chengdu, 30°35'52.8"N 104°04'23.1"E)

Figure 5. Trumpet structure [3].
(G4215 & Dongshan Blvd 2 Section Intersection, Chengdu, 30°24'57.3"N 104°07'44.5"E)

DOI: 10.54254/2755-2721/6/20230523

Then, the data are labelled as the different structures manually and stored in different directories. The data are normalized, whitened, and enhanced before being organized into training and test groups. Batch whitening has been shown to improve performance in discriminative scenarios [8]. In addition, data enhancement has shown a dramatic increase in the accuracy of CNN networks [8-9]. Figure 6 shows the process of data preprocessing.

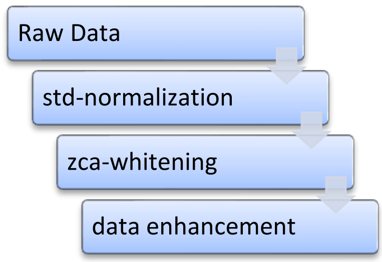

Figure 6. Data preprocessing.

As deep learning requires a lot of data, while intersection structures such as Stack only appears in downtown areas, thus being low in numbers, data enhancement is needed to ensure data is thoroughly used for better accuracy. In the paper, each data is:

Rotated between 0~90 degrees;
Width and Height shifted for 0~0.2;
Zoomed 0~0.2 times;
Horizontally flipped, vertically flipped or neither.

For training data, each image for training is processed to ensure generating 20 images(3500 in total), while test data uses original images(55 in total). The formulas for rotation conversion are shown as formulas(1) and (2).

$$\hat{X} = x * \cos \alpha + y * \sin \alpha \tag{1}$$

$$\hat{Y} = x * \cos \alpha - x * \sin \alpha \tag{2}$$

($\bar{X}$, $\bar{Y}$) is the position of the pixel after processed; while (x, y) is the position before processed. α being the rotation angle.

Before training, the data RGB values are reduced to (0,1) for min-max scaling. The formula is:

$$X = \frac{x - x_{min}}{x_{max} - x_{min}} \quad Y = X * (mx - mi) + mi \tag{3}$$

X is the middle value after scaling, and the x is the original value, x_{min} is the smallest value in the original data, whereas x_{max} is the biggest. Y is the final result, mx being the biggest in a designated range, whereas mi is the smallest [10].

3.2. *Model architecture*

Convolutional neural network, or CNN model, is a feedforward neural network that has high performance in large image processing. CNN networks can learn to classify features of images, and each layer is calculated by the local region of the upper layer, sharing the convolution kernel of weight. In a CNN, the convolution calculation is performed by a sliding mechanism and convolution kernel. The elements of a convolution kernel matrix and the elements of a covered area are multiplied and accumulated. Each convolution kernel is a feature extractor, and the convolution operations at all locations in training images share the weight of the convolution kernel [11].

3.2.1. LeNet-5 model. LeNet-5 is a classic structure of CNN and is mainly used to recognize handwriting. Figure 7 shows the structure of LeNet-5 [12].

DOI: 10.54254/2755-2721/6/20230523

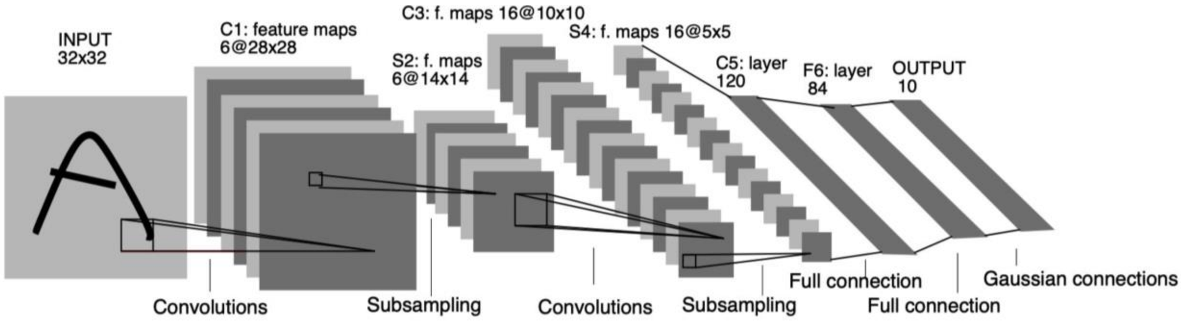

Figure 7. Structure of LeNet-5[13].

The LeNet-5 is built in this paper using Keras and tensorflow-gpu as the backend. The LeNet-5 of this paper consists of two convolutional 2D layers, each having a 5 by 5 kernel, before sending to 500 units, 'relu' activated DNN layer. Finally, the data is classified into five groups.

3.2.2. AlexNet model. AlexNet consists of 650000 neurons and 6 billion parameters. There are three DNN layers in the end, and the dropout function is used to reduce overfitting. Figure 8 shows the model of AlexNet [12].

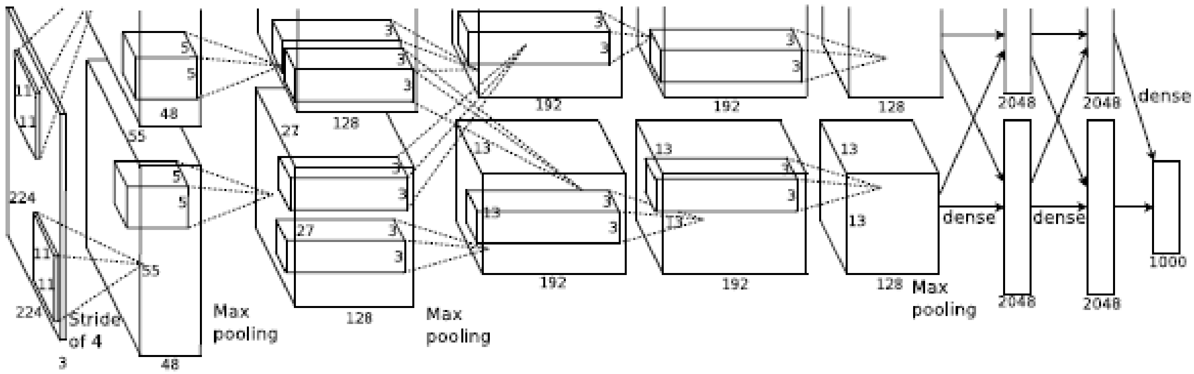

Figure 8. The architecture of AlexNet [13].

In AlexNet, most activation functions are 'relu', as it can descend the gradient faster than standard 'tanh' activation functions. In AlexNet, the dropout layer is set as 0.5, however, to reduce overfitting, the layer is set as 0.8 in this paper.

3.2.3. ResNet model. ResNet utilizes 'Shortcut connection' to reduce 'Degradation' to reduce the difficulties in training deep neural networks, the deepest neural networks reaching 1000 layers. He and Zhang discovered that as the depth increases, the accuracy reduces, which is abnormal [14]. To fix the problem, they added a 'Shortcut connection' to the model, significantly increasing the accuracy and speed of training while maintaining deep networks. Figure 9 shows the architecture of ResNet.

DOI: 10.54254/2755-2721/6/20230523

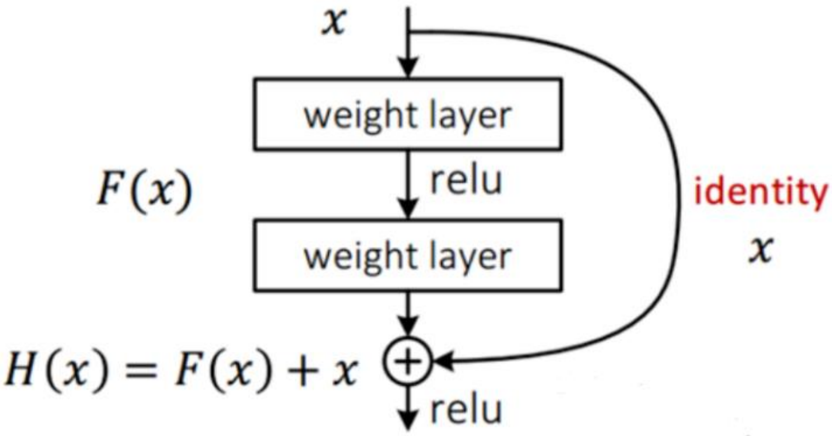

Figure 9. The architecture of ResNet [14].

In this paper, a ResNet of depth 50 is used, and the backend is tensorflow-gpu. ResNet has a high depth and takes a considerable amount of time to train, so the version with 50 layers is selected for better accuracy and less time consumption. The selection of Tensorflow-Gpu is to match the backend of other models, ensuring that all models have similar performance.

3.3. Evaluation

The evaluation is calculated by formula (4):

$$A = \frac{n}{N} \tag{4}$$

A is the accuracy, n is the number of correct predictions, and N is the total number of test samples. The time of training of different samples are also evaluated.

4. Training and results

In the paper, the resolution of all test and train samples is reduced to 400*225 in case of OOM(Out of Memory Error). The epochs are optimized for each model, and ‘sgd’ is chosen as the loss optimizer for better accuracy and less overfitting. Table 1 shows the accuracy and time of different models.

Table 1. The results of different models.

NAME	EPOCH	ACCURACY	VALIDATION-ACCURACY	TIME
ALEXNET	90	0.89	0.98	3:06
RESNET-50	150	0.75	0.96	33:47
LENET-5	95	0.94	0.75	2:59

4.1. LeNet-5

Of all the models, LeNet-5 took the least time(2 minutes and 59 seconds), but is the lowest in validation accuracy(54%). However, it has a high accuracy(100%) and a low epoch(30). This is an indication of overfitting and an unsuitable optimizer. By changing the optimizer from ’rmsprop’ to ‘sgd’, and adding another additional dropout layer, the validation accuracy has increased to 0.75. Table 2 shows the difference of the LeNet-5 results.

Table 2. The results of LeNet-5 ‘sgd’ and ‘rmsprop’.

NAME	EPOCH	ACCURACY	VALIDATION-ACCURACY	TIME
LENET-5(SGD)	95	0.94	0.75	2:59
LENET-5(RMS)	30	1	0.54	1:06

DOI: 10.54254/2755-2721/6/20230523

Figure 10 shows the difference in accuracy by the increase in epochs of the two variants of LeNet-5 model.

(Note: In figure 10, X coordinate stands for epochs, and Y coordinate stands for accuracy.)

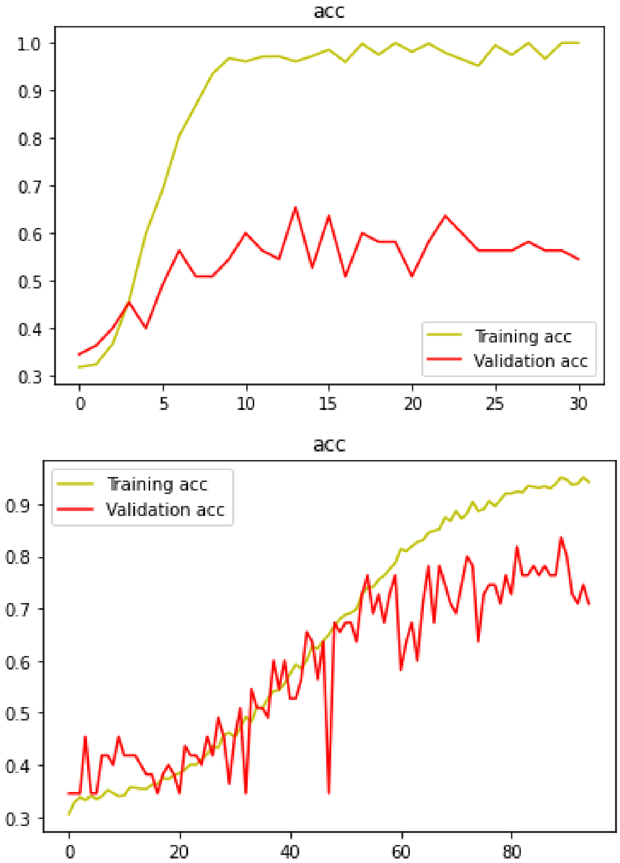

Figure 10. (Left) accuracy curve of LeNet-5 ‘rmsprop’ and (Right) accuracy curve of LeNet-5 ‘sgd’.

4.2. AlexNet

LeNet-5 is limited in processing complex data such as intersections. Thus, a more complex version of CNN, AlexNet is used. AlexNet is nearly perfect for processing the data, and reaches an accuracy at 0.98 with a low time consumption. However, the accuracy of training data is slightly lower than validation data, which may be a result of over preprocessing or a higher dropout rate. By reducing the dropout rate from 0.8 to 0.7, the accuracy will increase to 0.99, while the validation data accuracy will decrease to 0.94. Thus, the reason of the symptom may be a result of both, but in order to maintain a high accuracy rate for test samples, the dropout rate of 0.8 is chosen. Table 3 shows the difference of the two trials.

Table 3. The results of AlexNet with different dropout rates.

NAME	EPOCH	ACCURACY	VALIDATION-ACCURACY	TIME
ALEXNET 0.8	90	0.89	0.98	3:06
ALEXNET 0.7	90	0.99	0.94	2:40

DOI: 10.54254/2755-2721/6/20230523

Figure 11 shows the difference in accuracy by the increase in epochs of the two trials of AlexNet model. Figure 12 shows the loss curve of AlexNet(dropout rate of 0.8).

Note: In figure 12, X coordinate stands for epochs, and Y coordinate stands for loss.

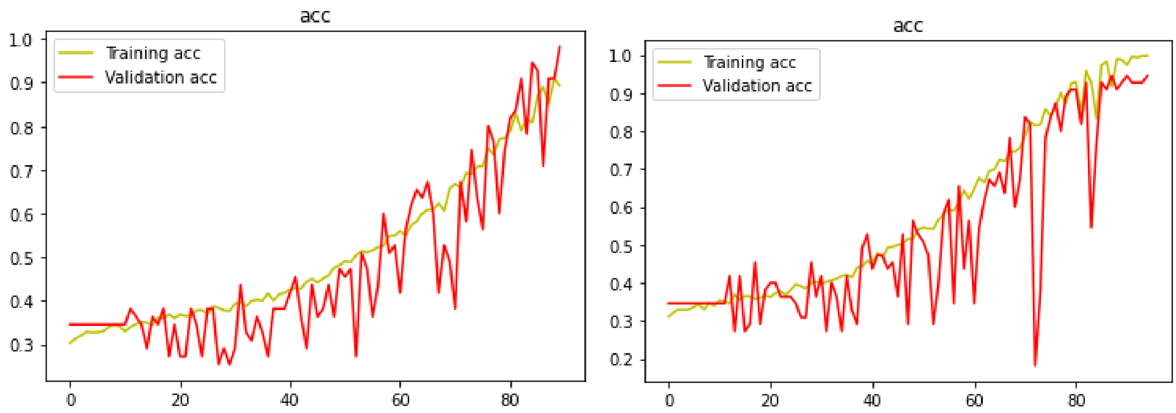

Figure 11. (Left) accuracy curve of AlexNet(dropout rate 0.8) and(Right) accuracy curve of AlexNet (dropout rate 0.7).

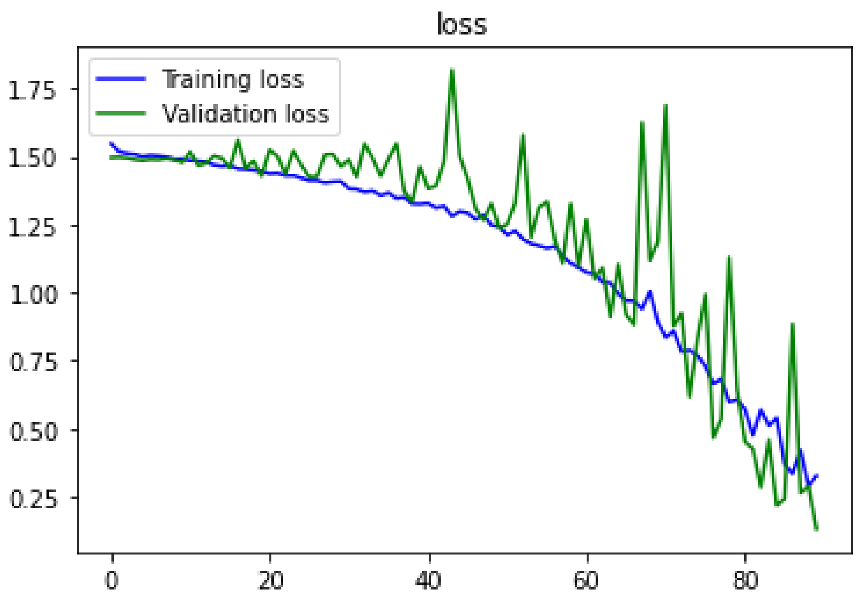

Figure 12. Loss curve of AlexNet(dropout rate 0.8).

4.3. ResNet-50

Of all the models used, ResNet-50 took the longest time (33 minutes compared to 2 minutes of AlexNet), and the validation accuracy is very close to AlexNet (96% compared to 98% of AlexNet). However, the accuracy of the training samples did not reach a high value(75%). Considering the fact that the ResNet-50 model has the highest depth, it may be sensitive to noise in data preprocessing. The evidence is that the model is hard to achieve higher accuracy after epoch 50, while the validation accuracy remains high until epoch 150(As shown in Figure 13). Though it took the longest time, it has a higher potential, as ResNet-50 and ResNet-150 only have a ten-second-difference in each epoch. However, as the depth increases, the time to decrease the gradient increases as well. Due to its high time consumption, the ResNet-150 is not tested. Figure 10 shows the difference in accuracy by the increase in epochs of ResNet.

DOI: 10.54254/2755-2721/6/20230523

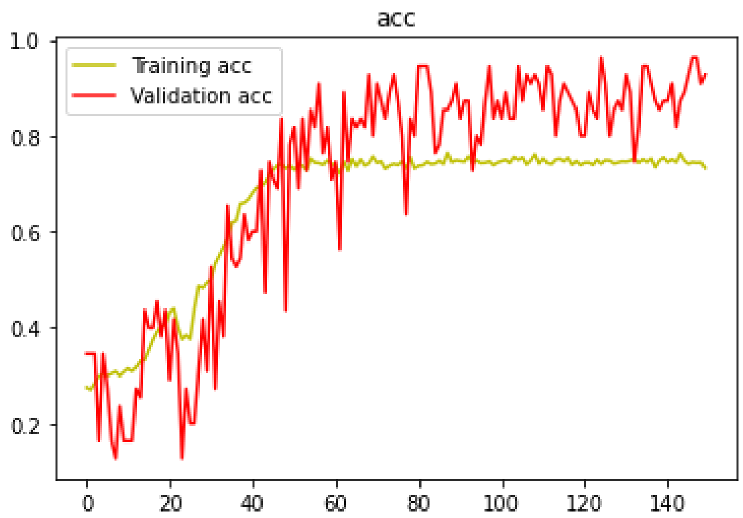

Figure 13. Accuracy curve of ResNet-50.

5. Conclusion

This paper utilizes different models of CNN to classify common overpass structures. The input only includes satellite image data and does not include any artificial labels. Because the road structure of the same class of overpass is relative stable, the model can be used in automated driving, which enables the vehicle to predict the road structures ahead without the need for accurate online roadmaps. There are some limitation in the paper, such as a lack of rare overpass structures in the dataset. When the structure of an overpass is strictly limited to landforms, the design of the structure may be hard to classify as a specific type of overpass. In future studies, road vector data can be added for classification of additional structures, and training data can be increased to enhance accuracy.

Acknowledgment

Firstly, I would like to show my deepest gratitude to my professor from University of British Columbia, who has provided me with valuable guidance in implementing artificial intelligence in this thesis. Further, I would like to thank all my friends and parents for their encouragement and support. Without all their enlightening instruction and impressive kindness, I could not have completed my thesis.

References

[1] Mackaness, W.; Edwards, G. July 2002. The importance of modelling pattern and structure in automated map generalisation. In Proceedings of the Joint ISPRS/ICA Workshop on Multi-Scale Representations of Spatial Data, Ottawa, ON, Canada, 7–8.

[2] Amap: http://ditu.amap.com/

[3] Google Maps: http://maps.google.com/

[4] Scheider, S., & Possin, J. (2012). Affordance-based individuation of junctions in Open Street Map. Journal of Spatial Information Science, (4), 31-56.

[5] Xu, Z., Meng, Y., Li, Z., & Li, M. (2011). Recognition of structures of typical road junctions based on directed attributed relational graph. Acta Geodaetica et Cartographica Sinica, 40(1), 125-131.

[6] Li, H., Hu, M., & Huang, Y. (2019). Automatic identification of overpass structures: a method of deep learning. ISPRS International Journal of Geo-Information, 8(9), 421.

DOI: 10.54254/2755-2721/6/20230523

[7] Haiwei, H. E., Haizhong, Q. I. A. N., Limin, X. I. E., & Peixiang, D. U. A. N. (2018). Interchange recognition method based on CNN. Acta Geodaetica et Cartographica Sinica, 47(3), 385.
[8] Huang, L., Yang, D., Lang, B., & Deng, J. (2018). Decorrelated batch normalization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 791-800).
[9] Zhang, F., Xu, X., & Qiao, Y. (2015, December). Deep classification of vehicle makers and models: The effectiveness of pre-training and data enhancement. In 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO) (pp. 231-236). IEEE.
[10] Namozov, A., & Im Cho, Y. (2018, September). An improvement for medical image analysis using data enhancement techniques in deep learning. In 2018 International Conference on Information and Communication Technology Robotics (ICT-ROBOT) (pp. 1-3). IEEE.
[11] Ciregan, D., Meier, U., & Schmidhuber, J. (2012, June). Multi-column deep neural networks for image classification. In 2012 IEEE conference on computer vision and pattern recognition (pp. 3642-3649). IEEE.
[12] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.
[13] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. Communications of the ACM, 60(6), 84-90.
[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. (2015). Deep Residual Learning for Image Recognition. arXiv 2015, arXiv:1512.03385.