Robustness Analysis of Traffic Sign Recognization based on ResNet

Kaiyao Li *

School of Taiyuan University of Technology, Shanxi, China

* Corresponding author email: likaiyao7347@link.tyut.edu.cn

Abstract. Autonomous driving has always been an important research topic and application task of artificial intelligence, which has attracted the attention of a large number of researchers. As an important component of the environmental perception module in autonomous driving tasks, traffic sign recognition can help drivers understand road information in a timely manner and avoid potentially dangerous driving operations. To this end, accurate recognition of traffic signs is crucial from both a strategic and a practical point of view. The early recognition technology of traffic signs is mainly based on the detection of color and shape, whose recognition accuracy is limited due to the fading and deformation of traffic signs. Numerous researchers have successfully used these deep learning-based object identification algorithms for traffic sign detection and recognition thanks to the development of the Faster R-CNN and YOLO series algorithms. However, we argue that few efforts focus on the recognition performance of the model in different scenarios, which is especially important in the process of autonomous driving. Based on this observation, this paper first constructs the ResNet model. Part of the image recognition accuracy predicted by the model reached 99%. Afterward, the robustness of the model is explored by simulating complex scenes by changing illumination and noise, and it is proved that the model has good generalization ability and practical application ability.

Keywords: Traffic Sign Classification; Deep Learning; ResNet; Autonomous Driving.

1. Introduction

With the continuous advancement of economic globalization in the twenty-first century, the global economy has developed rapidly, and the number of automobiles is increasing rapidly. Although the popularity of vehicles has greatly facilitated our ability to travel, it has also unavoidably led to a number of problems, especially traffic accidents. To assist drivers when driving, countries have begun to aggressively develop intelligent transportation and driverless technology in recent years. The goal of automatic driving is to obtain external information automatically and analyze this information in real time to make correct driving operations. As one of the key points of environmental perception in autonomous driving tasks, recognizing traffic signs can provide road information and driving precautions for drivers. To this end, accurate recognition of traffic signs is crucial from both a strategic and a practical point of view.

In fact, scholars from all over the world have long studied traffic sign recognition. The early approaches for detecting and identifying traffic signs may be separated into those that use color cues and those that use form features. The former is principally supported by the fact that traffic signs are composed of red, yellow, blue, and other hues, and that the majority of color distributions of traffic signs of the same kind are the same. A traffic sign identification technique based on pixel-level threshold segmentation was proposed by Janssen et al. in the RGB color space. Paclik et al. translated the image to HSI space and chose the hue threshold to acquire the necessary color since RGB color space is light-sensitive. The traffic signs can finally be detected by thresholding the saturation and intensity. However, since the color drop-off and fading of traffic signs will affect the detection methods based on color features, some works exploit different shapes among various traffic signs, which detect traffic signs by extracting edge information and analyzing their shapes and categories. Garc-Garrido et al. proposed a method for detecting traffic signs using the Hough transform of an image contour and discovered that the application effect is good. Belaroussi et al. achieved good detection of triangular traffic signs by detecting corner vertex and corner bisector [1]. Although the

detection method based on shape features can avoid color-shedding interference, the detection effect is greatly reduced when traffic signs are deformed or the detection angle changes.

Numerous researchers have successfully used these deep learning-based target detection algorithms for traffic sign identification and recognition thanks to the improvements of the Faster R-CNN and YOLO series algorithms. Luo et al. used a multi-task convolutional neural network to detect traffic signs and designed a multi-task loss function to ensure the detection process's stability. Cheng et al. proposed a Faster R-CNN traffic sign detection method based on local background for small target detection. The RPN network was used to extract candidate areas in order to classify local background information, and the local background was set to automatically extract classified information, after being tested using data sets of open traffic signs, the result was favorable [2]. To make it simpler to extract regions of interest, Zhu et al. offer a text-based traffic sign recognition approach that first uses a complete convolution neural network to produce candidate traffic sign areas [3]. A fast neural network is then used to identify the text retrieved from the areas of interest. The multi-scale text detection problem is resolved, the search area for text detection is reduced, the traffic sign text is removed, and the approach fully exploits the peculiarities of traffic signs. Following that, He et al. proposed the residual neural network as an innovative solution to the problems of gradient disappearance, gradient explosion, and degradation caused by increasing network depth. The results of the experiments demonstrated that the improved detection algorithm produced more accurate and robust detection results.

Although the existing research has greatly improved the accuracy of traffic sign recognition, few work has focused on discussing the robustness of the model, that is, the change of recognition performance in different scenarios, which is in the actual autonomous driving with changing scenarios. It is extremely important in tasks such as scene brightness, noise generated during image transmission and segmentation, etc. In response to the above problems, this paper builds a traffic sign recognition model based on ResNet and analyzes the recognition accuracy of the model in different scenarios in detail. Specifically, Section 2 will present the method's implementation specifics. We provide the experimental findings in Section 3 and talk about current issues with traffic sign recognition and potential future remedies in Section 4.

2. Methods

The traditional deep convolutional neural network, as we know, can add many layers to make the network deeper, with each layer seeing different features. However, if we want to build a high-accuracy network, we cannot simply pile the network deep. When the network is deep enough, the gradient will vanish and explode. When the weights are randomly initialized, the solution at first is to find a moderate value. The second solution is to include some middle normalization so that the output between each layer can be checked. However, as the network grows deeper, the training accuracy deteriorates, and both the training and testing errors become extremely large. As a result, the residual neural network is proposed.

2.1 Design Innovation of ResNet

The use of skip connections in the construction of a residual structure, which enables the network to penetrate further and enhance its performance, is the major innovation of ResNet [4]. The neural network degrades as the number of layers rises, meaning the deep-level network is inferior to the shallow-level network in performance. Additionally, over-fitting is not the reason of the deterioration gap because it is obvious in the training set. As a consequence, the deeper network shouldn't have the degradation issue with inferior performance than the shallow network if the neural network can easily achieve equivalent mapping between levels, that is, the input of a block equals the output of this block. He and colleagues came up with a skip connection structure as a consequence, which improved the network's identity mapping capabilities and increased the depth and performance of the network.

As shown in Figure 1, there are two types of mapping proposed in ResNet: identity mapping (the curve marked with an X on the right) and residual mapping (the residual refers to the F(x) part). f (x)+x is the final result. F(x)+x can be realized using a feedforward neural network with "shortcut connections." Shortcut connections are those that bypass one or more layers. The figure's "weight layer" refers to the convolution operation. If the network has been optimized, keep deepening it until the residual mapping is zero and only identity mapping remains. Theoretically, the network will always be in an ideal state if done this manner, and network performance won't suffer as depth rises.

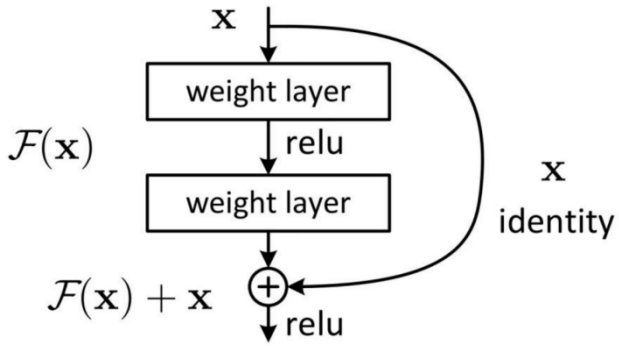
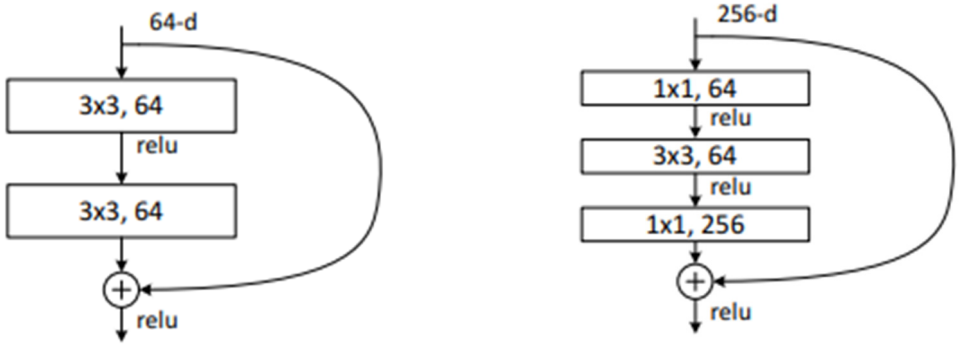


Fig 1. Framework of Residual mapping in ResNet

The residual block is made up of cascaded convolution layers and shortcut connections. After accumulating the output values of the two layers, the residual block output is obtained via the ReLU activation layer. A deeper network can be realized by connecting multiple residual blocks in series. As shown in Figure 2, the residual block can be designed in two ways. ResNet-18/34, a shallow network, belongs on the left of Figure 2, whereas ResNet-50/101/152, a deep network, belongs on the right. Utilizing this technique aims to reduce the number of parameters.



(a) Residual block in ResNet 18/34 (b) Residual block in ResNet 50/101/152

Fig 2. Residual block for different networks

2.2 Network Structure of ResNet

The net structure of ResNet has three main parts: pre-processing, conv_x layer and final full connection layer. Every conv_x layer is composed of several resblocks. Taking the most representative ResNet-50 as an example, its architecture is described in detail as following:

(1) Input layer: the image size is 224*224*3.

(2) conv1+BatchNorm+Scale+ReLU in convolution layer: 64 7*7 filter are used, with stride of 2, padding of 3 and output of 112*112*64, and 64 feature maps.

(3) conv2_x of convolution layer: the output is 56 * 56 * 256,256 feature maps. Maximum pool layer: filter is 3*3, stride is 2, padding is 0, output is 56*56*64, and 64 feature maps. There are three consecutive residual blocks, each with three layers of convolution; the convolution kernel size is 1*1, 3*3, 1*1, and the number of feature maps is 64, 64, and 256, respectively. Convolution+ BatchNorm

+Scale+relu is the first and second layers of convolution, and convolution+batch norm+ is the third layer. The first residual block's identity mapping requires a convolution kernel size of 1*1+BatchNorm+Scale operation, so that its output is adjusted to 56*56*256, which is convenient for Eltwise operation. Each residual block is followed by an Eltwise+ReLU operation.

(4) conv3_x of convolution layer: the output is 28 * 28 * 512,512 feature maps. There are four consecutive residual blocks, each of which contains three layers of convolution, the kernel size of convolution is 1*1, 3*3, 1*1, and the number of feature maps is 128, 128, and 512, respectively. The first and second layers of convolution are Convolution+BatchNorm+Scale+relu, and the third layer of convolution is convolution+batch norm+scale. Identity mapping of the first residual block needs convolution kernel size of 1*1+BatchNorm+Scale operation, so that its output is adjusted to 28*28*512, which is convenient for Eltwise operation. Eltwise+ReLU operation is performed after each residual block.

(5) Convolution layer conv4 x: the output is 14*14*1024, 1024 feature maps. There are six successive residual blocks, each with three layers of convolution. Convolution kernel sizes are 1*1, 3*3, and 1*1, and the number of feature maps is 256, 256, and 1024, respectively. Convolution+ BatchNorm+ Scale+relu is the first and second layers of convolution, and convolution+batch norm+scale is the third layer of convolution. The first residual block's identity mapping requires a convolution kernel size of 1*1+BatchNorm+Scale operation to adjust its output to 14*14*1024, which is convenient for Eltwise operation. Each residual block is followed by an Eltwise+ReLU operation.

(6) conv5 x convolution layer output: 7*7*2048, 2048 feature maps. There are three successive residual blocks, each with three layers of convolution. Convolution kernel sizes are 1*1, 3*3, and 1*1, and the number of feature maps is 512, 512, and 2048, respectively. Convolution+ BatchNorm+ Scale+relu is the first and second layers of convolution, and convolution+batch norm+scale is the third layer of convolution. The first residual block's identity mapping requires a convolution kernel size of 1*1+BatchNorm+Scale operation, so that its output is adjusted to 7*7*2048, which is convenient for Eltwise operation. Each residual block is followed by an Eltwise+ReLU operation.

(7) Average pool layer: filter 7*7, stride 1, padding 0, and output 1*11*2048, 2048 feature maps.

(8) Fully connected layer: the output is 1000 neurons or 1000 feature maps.

(9) Softmax output layer: output the classification result to determine which of the 1000 possible categories it is [5].

2.3 Loss Function

Since the original ResNet was designed for classification, the loss function of ResNet is the cross-entropy loss, which is used to calculate the softmax cross entropy between predicted results and ground truth. The distribution error between the input probability (calculated by the softmax function) and the objects with mutually exclusive categories is measured using cross entropy loss. The precise formula is as follows:

$$\ell(x_i, c) = -\log\left(\frac{\exp(x_i[c])}{\sum_j \exp(x_i[j])}\right) = -x_i[c] + \log\left(\sum_j \exp(x_i[j])\right) \quad (1)$$

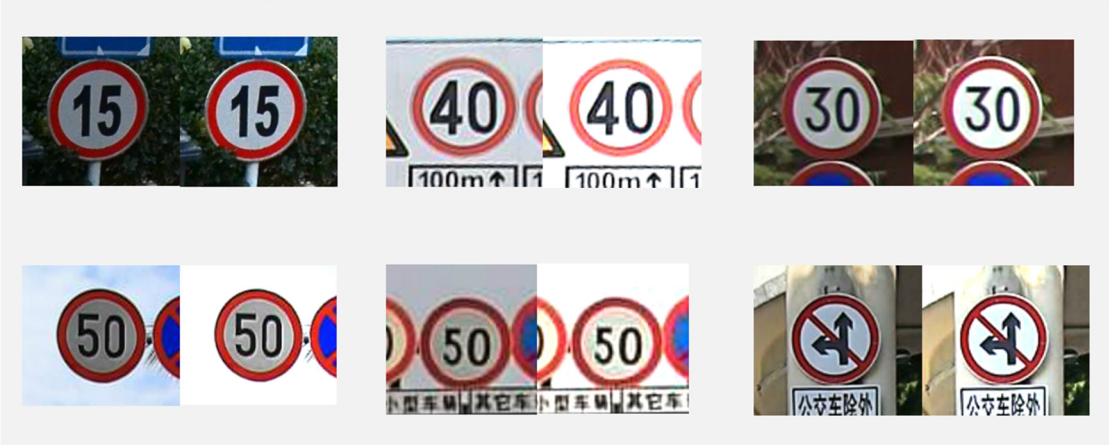
The output of the neural network's last layer serves as the first parameter in this situation. The size of a batch is [batchsize, num classes], and the size of a single sample is [num classes]. The second parameter denotes the actual label, whose size is the same as the one mentioned before [6].

3. Experiments

3.1 Dataset

In this study, we make use of an open-source data set from the Internet, which contains 312 images of speed limit 40, 194 images of speed limit 60, 446 images of motor vehicle driving, 156 images of pedestrian attention, 324 images of no parking, and 162 images of no driving. All of the images have a resolution of around 160. The deviation difference is used to adjust the brightness of some images

during the image preprocessing stage. Figure 3(a) depicts the contrast effect. The mean, median, and bilateral mixed denoising methods are used to reduce image noise, and some images clearly benefit. Figure 3(b) depicts the contrast effect. Following that, the data set is divided by random function, with a 7:2:1 ratio of training, verification, and test sets.



(a) Contrast effect before/after brightness enhancement

(b) Contrast effect before/after reducing noise

Fig 3. Original images of the dataset

3.2 Evaluation Indicators

The loss function's value can serve as a gauge of how well the network worked during model training. For the task of image classification, the accuracy can more accurately represent the final classification result, which can be calculated as:

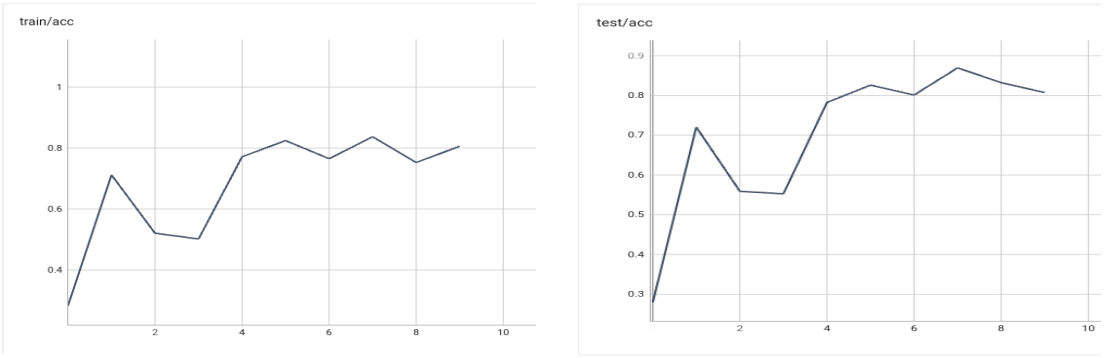
$$acc = \frac{ture\_positive+ture\_negative}{ture\_positive+ture\_negative+false\_positive+false\_negative} \quad (2)$$

3.3 Experiment Settings

The pre-trained ResNet-50 model is utilized to speed up the convergence of the entire model during the model training phase. The entire dataset is trained for 10 epochs with an initial learning rate set at 0.01. We adopt the Momentum as optimizer, whose main idea is to smooth the network parameters using a weighted average method similar to moving index so that the gradient's swing amplitude becomes smaller.

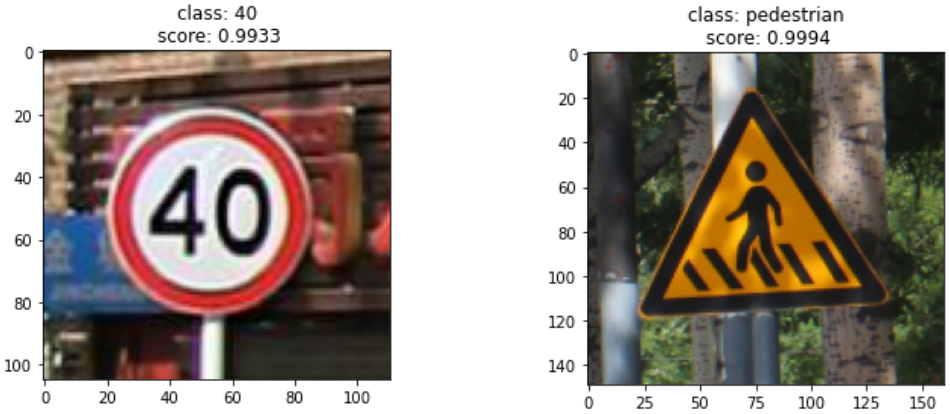
3.4 Experimental Results

Through the early stage of data set construction, image preprocessing and model training, we finally got a good correct rate, and the correlation curve is shown in the following Figure 4. The model's training curve shows that the model's training yielded good results, with the training accuracy of some images reaching 99%. We also visualize the recognition effect for some traffic signs in Figure 5.



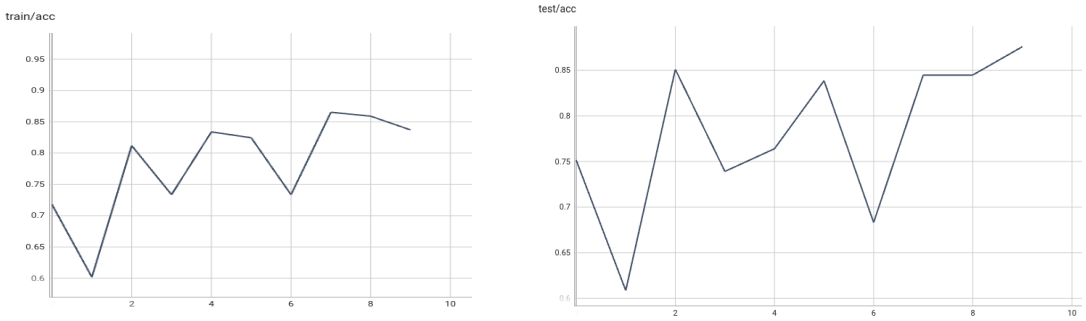
(a) Accuracy on training set (b) Accuracy on test set

Fig 4. Recognization accuracy of traffic signs



(a) Accuracy of 40 speed limit (b) Accuracy of pay attention pedestrian

Fig 5. Visualization of recognization effect of different traffic signs



(c) Accuracy on training set (d) Accuracy on test set

Fig 6. Recognization accuracy for different noise and brightness

However, as previously stated, this model does not know whether its generalization ability is strong enough to achieve such accuracy in complex scenes, so this paper focuses on two major aspects. The first consideration is the brightness. In the real world, street signs are frequently in the shade of

trees or in the shadow of tall buildings, and changes in light and shadow in cities are complicated, affecting machine recognition and prediction of traffic signs. The second factor is noise, which is common during the image transmission process, particularly in automatic driving. Noise will appear during the image segmentation process and the vibration of various instruments while driving, and the noise will also affect the model's prediction accuracy. The robustness of the model is demonstrated in the following experiment by varying the brightness of traffic signs and adding noise to observe the change in training accuracy of the model. Figure 6 depicts the model training effect. It can be seen from the figure that the model does not fluctuate too much, which proves that this model has strong generalization ability and has practical application value.

4. Discussion

Deep learning is one of the most important technologies in the development of self-driving cars. After decades of development, the machine learning technology, which forms new laws, judges, guides, and predicts unknown phenomena based on laws, has finally ushered in the technological explosion in the field of deep learning through the analysis, induction, and deduction of objective phenomena. Deep learning has effectively solved all subtasks in the traditional computer vision field over the last ten years, and the application from laboratory to industry has been truly realized. The increasing depth of the network and the complexity of the network structure not only show that computing power has greatly improved, but also demonstrate people's boundless imagination and superb insight. (1) The main work of this paper is to predict traffic signs using the ResNet-50 neural network, and good results are obtained. (2) To test the model's generalization ability, adjust the brightness of the input image and add noise to simulate the real world. [7] In comparison, the neural network outperforms the traditional algorithm in terms of performance and robustness.

The main focus of this paper is traffic sign recognition in a complex environment. Although the model has demonstrated good accuracy and generalization ability, there are many factors in the natural scene, making it difficult to recognize traffic signs. Traffic sign detection and recognition in intelligent transportation systems require high accuracy and real-time response. As a result, based on these factors, there is still much room for improvement in some of the traffic sign recognition links in this paper. The following are the primary work directions for the future: (1) In intelligent transportation systems, just the identification of traffic signs is too limiting, and the particular application scenario is unreliable. In addition, since traffic sign detection takes place in a live, natural environment, we can think about incorporating a detection and tracking algorithm to boost the effectiveness of subsequent recognition. (2) In the future, we should add more categories to the data set to enrich and expand the sample types in order to further increase the robustness of the data set [8]. (3) In order to decrease computation time and memory usage and increase detection speed for traffic sign identification, we must keep improving the structure of our convolutional neural network while adding additional features [9-10].

5. Conclusion

In this study, we present a ResNet-based traffic sign identification approach with the goal of evaluating and discussing the model's performance in various circumstances. Specifically, we first constructed a ResNet-based traffic sign recognition model, which achieved a recognition accuracy of up to 99%. Second, by simulating complex scenes by changing illumination and noise, we tested the variation of model recognition accuracy respectively. Numerous experimental findings demonstrate the model used in this paper's construction to have strong generalizability and applicability.

References

[1] Cheng Y M, Pan Q, Zhang H C, et al. Computer Intelligent Image Recognition Algorithm[J]. Computer Applications, 2004.

[2] Zhu Yanzhao and Yan Wei Qi. Traffic sign recognition based on deep learning[J]. Multimedia Tools and Applications, 2022, 81(13) : 17779-17791.

[3] Zhu Y, Baratoff G, Kohler T. SYSTEM AND METHOD FOR VEHICLE DETECTION AND TRACKING [J]. 2008.

[4] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[J]. IEEE, 2016.

[5] Yandong L I, Hao Z, Lei H. Survey of convolutional neural network[J]. Journal of Computer Applications, 2016.

[6] Liu Hongwei and Li Xiang and Gong Wenyin. Research on Detection and Recognition of Traffic Signs Based on Convolutional Neural Networks[J]. International Journal of Swarm Intelligence Research (IJSIR), 2022, 13(1) : 1-19.

[7] Zhang H. Application of Computer Image Recognition Technology and Analysis of Details[C]// 2019 International Conference on Information Science, Medical and Health Informatics. 2020.

[8] Karthika R. and Parameswaran Latha. A Novel Convolutional Neural Network Based Architecture for Object Detection and Recognition with an Application to Traffic Sign Recognition from Road Scenes[J]. Pattern Recognition and Image Analysis, 2022, 32(2) : 351-362.

[9] Liu Jingjing et al. How Can Sustainable Public Transport Be Improved? A Traffic Sign Recognition Approach Using Convolutional Neural Network[J]. Energies, 2022, 15(19) : 7386-7386.

[10] Z. Zeng, J. Wang, B. Chen, T. Dai, and S.-T. Xia, "Pyramid Hybrid Pooling Quantization for Efficient Fine-Grained Image Retrieval," arXiv preprint arXiv:2109.05206, 2021.