

Convolution Neural Network-Based System for Detecting Cracks on Exterior Wall

Xiong Hui^{1,a,*}, Li Geng^{1,b}

¹*Guangdong Jinglong Construction Group Co., LTD. Hanjing Road 30, Guangzhou, P.R. China*

^a *xhui@163.com*, ^b *ligeng@zgjinglong.com*

**Corresponding author*

Keywords: convolutional neural network, crack detection, image processing

Abstract: Based on the mathematical model of convolutional neural network, the database is built by taking images of external walls by unmanned aerial vehicle; a crack detection system for external wall and its facing material is established by software and hardware. The system can effectively identify the severity, general or slight damage degree of external wall cracks, and the effective identification rate is 86%, 91% and 97% respectively.

1. Introduction

The main purpose of this paper is to provide an effective system for the detection of cracks in tall buildings, so that the complex, dangerous and costly routine work can be simplified and saved through information technology. A small UAV (unmanned aerial vehicle) equipped with intelligent operation system is proposed to identify the damage degree of the veneer material in earthquake or normal times, so as to reduce the number of climbs for workers, carry out rapid detection, identify the location of potentially dangerous veneer material, and evaluate the degradation degree of the veneer material. In order to achieve the above purpose, based on machine learning as the main method, deep learning as the core, by using CNN (convolutional neural network) technology and the identification function implemented on the embedded development board, it is simple and easy to identify the damage degree of the veneer material on exterior wall.

2. Research Method

In order to classify and analyze the data collected by UAV, it is necessary to establish the environment of Tensor flow^[1] in the embedded development board (Jetson TX2) and the computer host at the same time, and use Keras^[2] as the foundation of erecting modules, then set up CNN model with Keras and use Tensor flow as the back-end for training. Before training, the data set should be divided into several categories (namely labelling) and the data have been classified, so as to classify photographs of similar deterioration into one category.

The CNN above is a model designed on the basis of quasi-neural network, which can change the computation of quasi-neural network by reducing the neuron and extracting features. Compared with the traditional image processing methods, CNN is able to process complex images and has the ability to learn, which can also be used for prediction. The present mainstream designs of CNN

include lenet-5, AlexNet, network-in-network and GoogLeNet, etc. Among them, AlexNet uses the maximum pool technique, imports the Rectified Linear Units (ReLU) mechanism, and makes neurons for the problem of gradient disappearing in nonlinear situation. AlexNet performs well in recognition accuracy and overall training speed when processing large amounts of data and retrieving more features. Therefore, we decided to adopt AlexNet as the main model to achieve the goal after some comparisons. Through the input of ImageNet's image database for training^[3], the model was able to distinguish 1,000 types of images from 1.2 million images as training samples and 50,000 as the test group. In the CNN model, the convolution program mainly uses the inner product of image and convolution kernel to extract the required features. Some applications can be found in the references [4-6].

As shown in figure 1, the inner product of the matrix (dimension 3 by 3) in the red box with the inner product of the convolution kernel of dimension 3 by 3 fills the corresponding position of the 2-by-2 matrix on the right of the equal sign, which is the characteristic value. The inner product of the same convolution kernel can be used to obtain other eigenvalues and fill in the matrix of the right side of the equal sign one by one. The eigenvalues obtained by the matrix record will be reserved to the next layer for subsequent calculations.

1	1	0	1	*	1	0	1	=	3	3
1	0	1	0		0	1	0		4	2
1	0	1	0		1	0	1			
0	1	1	0							
Image					Convolutional kernel					

Figure 1 Convolution operation example (the 2-by-2 matrix on the right of the equal sign is the eigenvalue extracted from the convolution kernel according to the original image)

After the convolution layer finds the features, it can be used as the input variable to carry out the general neural network classification model training. AlexNet uses dual-GPU (Graphics Processing Unit) to train the neural network, so the entire network is distributed to two GPUs to execute, and through the features of the current GPU architecture, one GPU is able to directly access the memory of another GPU without accessing the memory of the host. This feature can be used to implement the parallel operations.

3. Image Preprocessing

The image data of cracks in external wall or decorative materials are collected and processed by UAV. First, an image processing environment is constructed, and the data is retrieved by 224×224. After that, the image is input to the PC host and the image is preprocessed. The reason why 224×224 is chosen as the treatment unit is based on the comprehensive consideration of budget quantity and detection rate after lots of operations. The purpose of image preprocessing is to convert the original RGB three-channel original image captured by the UAV into the form of simple color, but still retain the important information of the original image, so as to enable the machine learning software system to capture its important features more effectively and avoid interference in learning due to the color difference of image RGB. The reason for this is that during the research process, many experiments showed that the effect of machine learning was not as expected. After detailed inspection, it was found that AlexNet mistook the color of wall tiles as one of the features to be learned, but the color of wall tiles was not the key feature in this study, so the image should be

preprocessed. The steps and algorithm of image preprocessing are as follows:

(1) Gray-level Transformation. Taking the formula

$$Gray = 0.299 \times R + 0.587 \times G + 0.114 \times B. \quad (1)$$

As shown in figure 2, the right figure is the schematic one after the graying process of the left figure by formula (1).

(2) Image binarization. Binarization is mainly used to convert the grayscale image into black-white image, and this program is mainly used to distinguish the part that the researcher is interested from the part that does not. By setting a gray threshold, the complex environment can be removed and only the portion showing the crack. The adaptive binarization function of software MATLAB (version 2018a) is used to select the appropriate threshold value for different images, and the ideal binarization result can be obtained. Figure 3 shows the adaptive binarization results of figure 2.



Figure 2 The original image and its grayscale processing results

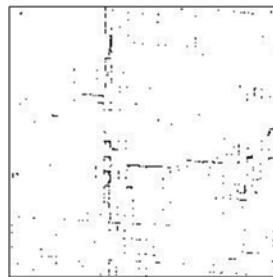


Figure 3 Adaptive binarization result

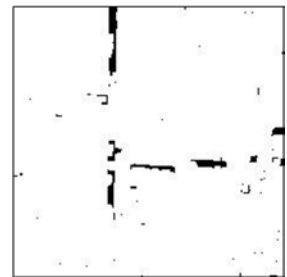


Figure 4 Result of open operation

(3) Morphological operation. Morphology is used in binarization images to highlight shapes or features (such as boundaries or fine lines) of images by mathematical processing. The most basic operations in morphology include dilation, erosion, opening and closing. In this paper, open operation is used mainly to enhance the contour of the crack and filter out some stray points. The results are shown in figure 4.

4. Data classification and parameter adjustment

4.1 Image Data Collection

The original image data is collected by the UAV. The unprocessed data is the whole image that has not been retrieved, as shown in figure 5.



Figure 5 Original image of crack in exterior wall



Figure 6 A schematic diagram



Figure 7 Four examples of cracks in exterior wall

The small red box is the schematic diagram of the range of retrieval to be processed. Before the original image is processed, the interference factors (such as windows, trees and other parts

irrelevant to the degradation of wall tiles) are deliberately excluded. Then, the tile crack part of the original image is extracted with a resolution of 224×224 , and hundreds of similar local images will be obtained. Figure 6 is one of them. The images of external wall cracks in this paper were collected in urban public buildings, such as libraries, gymnasiums, commercial buildings, etc. Figure 7 is a few examples of the cracks.

4.2 Data Classification

After all the images have been preprocessed, they are sorted by volume label, such as severe, general and slight. An example of an image category volume icon is shown in figure 8. After setting the volume label, place the data in the training and test folders respectively for subsequent access. Considering the powerful image processing capability of the computing software MATLAB (version 2018a), we can consider using MATLAB for bulk preprocessing. After the preprocessing, CNN would be built and the development environment is needed to install Python, Tensorflow and Keras, which is the most common combination at present. In order to accelerate the whole training process, GPU is used on PC side for training. Finally AlexNet as the core network of training is chosen to use.

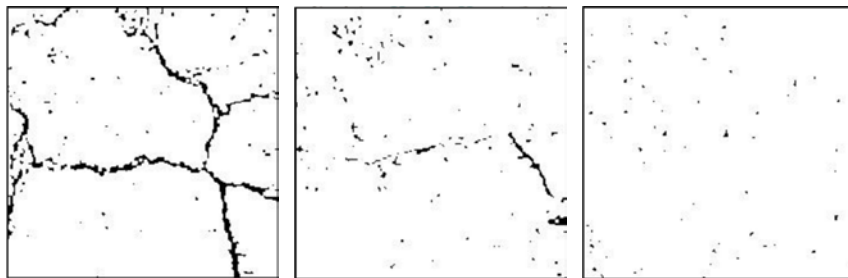


Figure 8 Volume labels are set to severe, general, and minor levels from left to right

4.3 Training and Adjustment

Train AlexNet and get the appropriate model by entering different data sets and calibration parameters. Basically, the accuracy curve and loss curve can be observed to determine whether the parameters are feasible. As shown in figure 9-left, in order to show the learning curve of the training set and the validation set, respectively marked with blue (accuracy) and orange (val_acc), the closer the value is to 1.0, the more correct the model training is. The loss curve of figure 9-right represents the error rate of the model, and the lower the value, the lower the probability of judging the error of the picture.

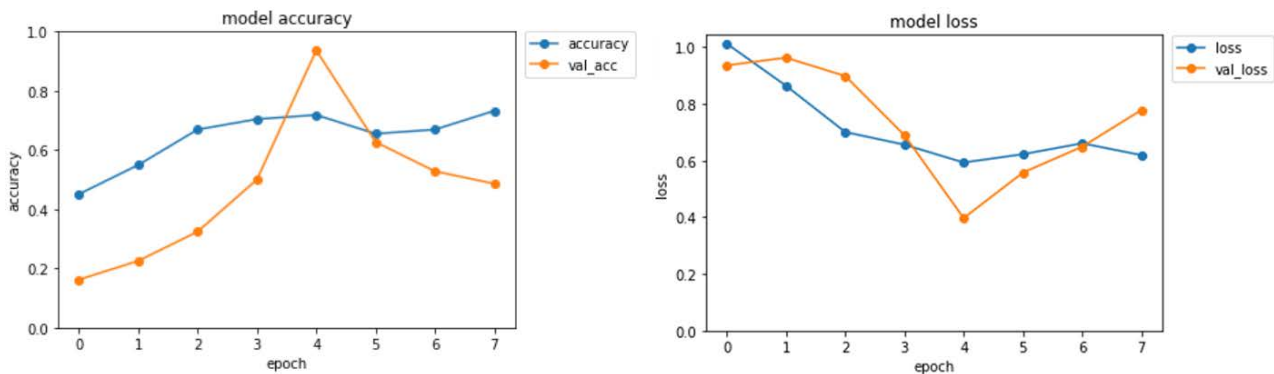


Figure 9 Accuracy learning curve(left) and loss curve(right)

The data of the verification set and training set are obtained from the image classification of the training set. Two observations can be made from the above curve. First, as shown in figure 10-left,

if the curve of the test set grows slowly, and there is no significant increase or jump (in the case of $\text{epoch} \geq 5$), it indicates over-fitting. The main reason is that the learning process is too long or the training sample amount is too small. Once this happens, one should try changing the data set, such as adding pictures or adjusting parameters. As shown in figure 10-right, the training set curve of the loss curve gradually decreased with the epoch increasing, the curve of the verification set did not show corresponding growth or bouncing phenomenon. This situation may be due to the wrong volume label of the image, which makes the model unable to learn correctly. You can go back to check whether there is classification error in the volume label of the image.

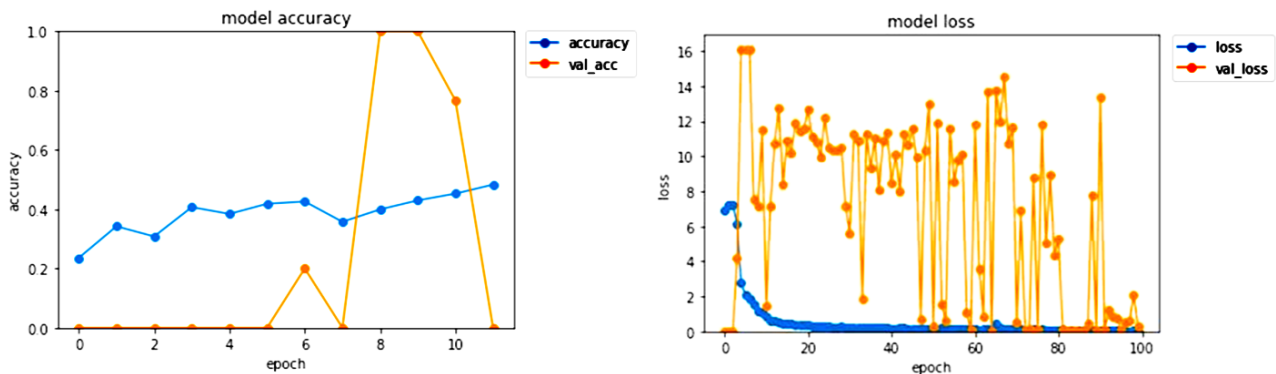


Figure 4.7: Over-fitting curve(left) and curve of uncorrectly marked volume(right)

Parameter adjustment is a subject that must be faced by training using neural network. The core parameter of our study is the learning rate, which can be adjusted by observing loss curve. In addition, the allocation proportion of training set and verification set can be adjusted to make the model learn more correctly.

5. Result of Image Processing

5.1 Execution Steps in Experiment

In order to test the effectiveness of the method, the execution steps in experiment are as follows:

- (1) Converting test data to 224×224 dimensions, regardless of any size;
- (2) Confirming the integrity of test data;
- (3) Identifying pictures;
- (4) Making statistics for verification results;
- (5) Comparing the error rate of each model.

5.2 Test by Batch

The test data is divided into 5 batches, which are labeled correctly. Each batch of test images is 20, so there are a total of 100 test data. The availability of the model was verified by statistical prediction of whether the results conform to the volume label and by observing the learning curve of the model. Table 5.1 shows the number of sheets of each batch test set category, which is verified by input of different types of test sets, thus avoiding the specificity of image selection. Among them, DataSet4 and DataSet5 have zero test sheets for specific categories, and the main intention is to confirm that the number of error sheets for specific categories is not high.

Table 1 Three test sets

Test set	DataSet1	DataSet2	DataSet3	DataSet4	DataSet5
slight	5	5	10	10	0
fair	5	10	5	0	10
serious	10	5	5	10	10

After preprocessed, the correct identification rate of three types of volume marks is effectively improved, and the comparison results are shown in table 2. Obviously, the identification rate of preprocessed images is much higher than that of unprocessed images.

Table 2 Comparison of identification rates of three types of test sets

Volume	After preprocessed	Unprocessed	Amplitude of growth
slight	97%	66%	46.97%
fair	91%	33%	175.76%
serious	86%	44%	95.45%

References

- [1] Tensorflow, <https://www.tensorflow.org/>, 1, Sept., 2018.
- [2] keras Documentaion, <https://keras.io/>, 1, Sept., 2018.
- [3] Vu H T, Huang C C. A multi-task convolutional neural network with spatial transform for parking space detection[C]. *IEEE International Conference on Image Processing. IEEE*, 2017:1762-1766.
- [4] Jinsheng X, Enyu L, Li Z, et al. Improved image super-resolution algorithm based on convolutional neural network [J]. *Acta optics*, 2017(3):96-104. (in Chinese)
- [5] Min Z, Zhenwei S, Huoping D. Convolutional neural network method for target classification of remote sensing image aircraft [J]. *Chinese journal of image and graphics*, 2017, 22(5):702-708. (in Chinese)
- [6] Yun G, Shunliang J, Xingmao Y, et al. Remote sensing image retrieval based on ImageNet pre-training convolutional neural network [J]. *Journal of wuhan university (information science edition)*, 2018, 43(1):67-73. (in Chinese)