ISBN 978-81-936279-9-0
10th Int'l Conference on Innovations in Engineering, Technology, Computers & Applied Sciences (IETCAS-18)
Kuala Lumpur (Malaysia) July 23-25, 2018

Ensemble Model using Multiple Image Pre-processing Method and CNN Model for LCD Panel Defect Classification

Minyoung Lee[1], Myungso Kim[1], Minjeong Ann[1] and Hongchul Lee[1]

[1]Korea University Information Network Lab

Abstract**:** *In this paper, we deal with the problem of classifying defects in defect images of LCD panels. Preprocessing techniques for images, individual model generating, and ensemble techniques are used to solve the problem. Since the image preprocessing method may have different effects depending on the applied field, various methods are applied to increase the effect. The final model uses the ensemble model to improve the performance of the model.*

Keywords: *convolutional neural network, ensemble model, image processing, classification*

1. Introduction

It is important to detect and classify defects that occur during the process in an LCD panel process. Detecting and classifying defects is most often done by a technician. This task requires a lot of time and manpower, and there is a difference in the accuracy of judgment depending on the skill level of the technician. If there is a model that automatically classifies defective patterns that may occur in the panel, it can perform classification operations with help of machine judgment regardless of skill level. Due to the nature of the LCD panel, pattern data, which can be called the background, occupies most of the data, and some parts show defects. A model capable of learning data including such a background pattern is needed. In the defect classification study of the LCD panel, research for removing the background pattern was preceded. A repetitive pattern is preferentially detected in the image, and the remaining background pattern is removed by leaving only the defect portion in such a manner as to remove the same image as the detected pattern. After the background pattern is removed, it is transformed into better data for learning in the machine learning or deep learning model [1]. Convolutional Neural Network model, which is most used for image learning, can be applied as a model for learning defect classification. In other researches, it has been developing models that use image data and Convolutional Neural Network in many fields, but they are experiencing difficulties in terms of data quality and model utilization. To solve the problems mentioned above, in this study, we apply various pre-processing methods to remove repeated patterns of LCD panel data, generate CNN models for the applied method, and then generate the ensemble model for defect classification. In Section 2, we describe the proposed technique and model, and in Chapter 3 we discuss the experiment and the results. Section 4 discusses experimental results. Chapter 5 discusses future research.

2. Proposed Model

2.1. Data Pre-process

In order to increase the learning efficiency of the data, a method of applying a deformation to the original image or removing the same appearing pattern can be utilized. An example of pre-processing in LCD panel is J. J. Lee[2] and K. M. Lee[3]. J. J. Lee[2] improved the performance by using the image pre-processing method to obtain repeated patterns and remove the patterns except the defective parts by recognizing the parts that appear different from the pattern as defects. K. M. Lee [3] presented a method of improving learning efficiency by reducing the difference in brightness that can occur in the image after black and white processing.

In this study, we used 4 data pre-processing techniques. First, the pattern is found in the Origin image, and the part having the same pattern is detected. The portion other than the portion detected by the pattern is assumed to be a defect, and the defect portion is emphasized. The type of pre-processing depends on the processing of the part detected by the pattern. Type-1 data removes all pattern parts and leaves only the defect part in the image. The type-2 data weakly adjusts all values of the detected portion of the pattern. In the type-3 data, the threshold value is applied stepwise to the part detected by the pattern, and the pattern part included in the same section converts the data having the same value. In addition, for all data, find the center of the defect and crop it. Using all of the data pre-processing techniques, a total of eight types of data sets are created from the original image data. Instead of using only the original data, we can extract various features through various pre-processing processes and use them for learning. Although the original image has all of the information in the data itself, in the case of images that contain repetitive patterns, such as the data in this study, viewing the entire original of the data may act as noise in the actual classification model. Fig. 1 shows the original image with grayscale applied and an example of each pre-processing method.

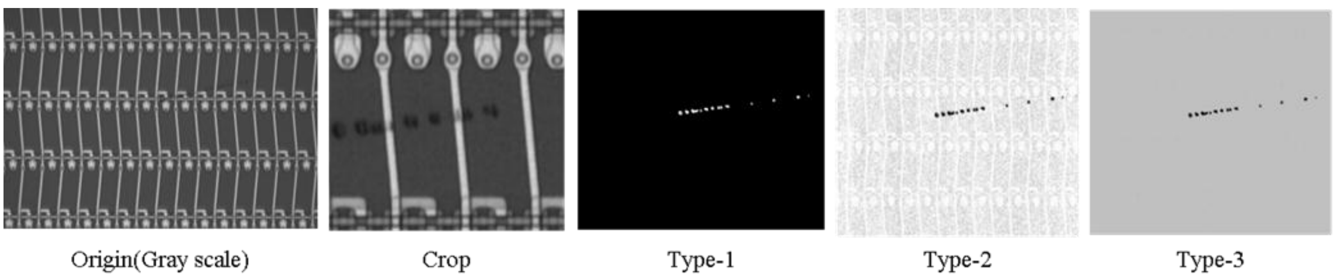

Fig. 1: Example of each pre-processing methods

2.2. Individual Models

A defect classification model is created using each data set generated according to the data pre-processing method mentioned in Section 2.1. The generated individual models are utilized in the final ensemble model. Utilizes the VGG-16 network architecture to create individual models. VGG Network is a model made by K. Simonyan and, A. Zisserman[4]. Among the proposed models, the best performing model is 16 convolutional, fully connected layer, and this model is called VGG-16 Network. The mode convolutional layer of the VGG Network is 3x3 and the pooling layer is 2x2. In recent studies, it is most commonly used for image feature extraction. The image size used in the model learning was 256x256, the learning rate was 0.001, and the SGD optimizer (decay: 1e-6, momentum = 0.9) was used. Through the completed learning model, we can classify the class of input data and obtain the prediction probability for each class. In this study, 8 data sets were created as mentioned in Section 2.1 and created individual models using each data set.

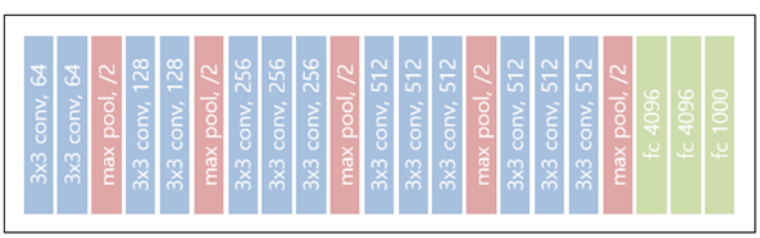

Fig. 2: VGG-16 Network Structure

2.3. Ensemble Model

The data set generated in Section 2.1 contains different features according to each pre-processing method. Since each model generated in Section 2.2 is a model that learns the features of each image set, we obtain the class classification probabilities of each model. The class classification probabilities obtained from individual models are used as inputs to the final class prediction ensemble model. The Ensemble Model applies a simple MLP model to prevent overfitting of certain models. Classify the class of defects using the predictive model obtained from each individual model. Fig. 3 shows the process from the original image to final ensemble model generation.

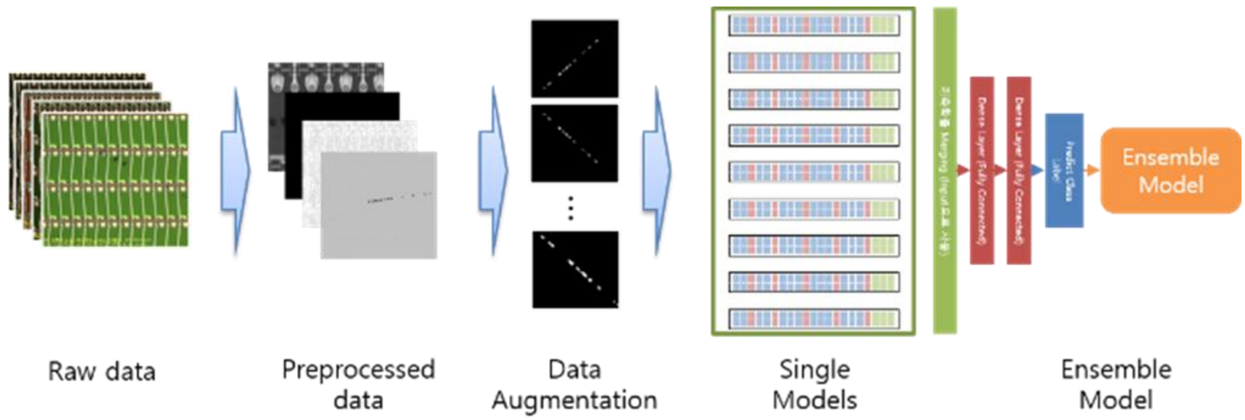

Fig. 3: Final model generation structure

3. Experiments and Results

Experimental environment was CPU: i5-6600, GPU: GTX 1080 TI, RAM: 32GB, all in the same environment. TABLE 1 shows the classification performance of each individual model. Dataset means dataset according to pre-processing method used, and train, test and valid set is same with 1680, 72, 476. All experiments were conducted using the same random seed to prevent performance differences due to the configuration of train and test set. To compare the effects of the same epoch experiment, all experiments were performed by 1000 epoch.

TABLE I: Results of Single Models

No	Dataset	Train Accuracy (%)	Test Accuracy (%)	Valid Accuracy (%)
1	Original(Grayscale)	79.4	76.5	74.8
2	Original(Grayscale) + Crop	83.4	81.2	79.4
3	Type-1	82.3	82.6	83.0
4	Type-1 + Crop	82.8	81.1	85.1
5	Type-2	76.8	82.82	84.9
6	Type-2 + Crop	83.3	84.9	87.8
7	Type-3	77.8	81.4	83.2
8	Type-3 + Crop	81.8	83.8	85.3

The ensemble model was compared while adjusting the number of ensemble models used. The results of the ensemble model are shown in TABLE 2.

TABLE II : Results of Ensemble Models

Ensemble Model	Train Accuracy	Test Accuracy	Valid Accuracy
1-8	91.1	89.7	90.8
1-6	90.1	91.4	90.1
1-4	90.5	92.5	88.7

4. Conclusions

As can be seen from TABLE 2, performance tends to improve as the number of ensemble models increases. The 1-8 ensemble model is model using the entire dataset, 1-6 ensemble model excluding type-3 and type-3 + crop datasets, 1-4 ensemble model exclude type2, type-2 + crop, type-3 and type-3 + crop data sets. As the information about the pre-processing type decreases, the performance of the ensemble model decreases. Also, the number of input models used in the ensemble model can be considered to affect performance. Comparing the results of TABLE1 and TABLE2 shows that the performance of the ensemble model is much better than the final performance of Table 1, and even though the performance of the individual model is not very good, the ensemble model can be used to obtain good results.

5. Future Research

The improvement in this study is that if you increase the number of models to be used as inputs to ensemble model, you can expect additional performance improvement. To increase the number of input models, we can further study the pre-processing method. In this study, we apply VGG-16 network to all individual models. We expect that application of appropriate network to each data set will help improve performance. In addition, we use the MLP model in ensemble model generation, but we expect to improve the performance of the ensemble model by using better ensemble techniques such as checkpoint ensemble [12].

6. Acknowledgements

This work was supported by the BK21 Plus (Big Data in Manufacturing and Logistics Systems, Korea University) and aim Systems.

7. References

[1] S. Y. Kim, et al. “TFT_LCD Defect Detection Using Multi-level Threshold and Probability Densify Function”, Korea Institute of Intelligent Systems, 2009. Vol 19. No.3

[2] J. J. Lee, et al. “A method of pattern removal based on perspective transformation for defect detection of TFT-LCD”, Journal of Korea Multimedia Society Vol 15.6 (2012): 784-793.
https://doi.org/10.9717/kmms.2012.15.6.784

[3] K. M. Lee, B, K, Park. “Hybrid Defect Inspection Method for TFT-LCD Panel”, Division of Electronic and Electrical Engineering (Cotrol program), 2008, 28p.

[4] K. Simonyan and, A. Zisserman “Very Deep Convolutional Networks for Large-Scale Image Recognition” arXiv:1409.1556v6 [cs.CV] 10 Apr. 2015

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks” Advances in Neural Information Processing Systems 25 (NIPS 2012)

[6] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition” arXiv:1512.03385v1 [cs.CV] 10 Dec. 2015

[7] E. Y. Lee, K. H. Park, "TFT-LCD Defect Blob Detection based on Sequential Defect Detection Method", Journal of the Korea Industrial Information Systems Research Vol. 20 No. 2, Apr. 2015: 73-83

https://doi.org/10.9723/jksiis.2011.16.2.073
https://doi.org/10.9723/jksiis.2015.20.2.073
https://doi.org/10.9723/jksiis.2014.19.2.073

[8] S. Yang, et al. "An Ensemble Classification Algorithm for Convolutional Neural Network based on AdaBoost", IEEE ICIS 2017, Wuhan, China, 2017.

https://doi.org/10.1109/ICIS.2017.7960026

[9] D. Weimer, et al. "Design of deep convolutional neural network architectures for automated feature extraction in industrial inspection", CIRP Annals-Manufacturing Technology 65, 417-420, 2016.

https://doi.org/10.1016/j.cirp.2016.04.072

[10] J. Masci, et al. "Steel defect classification with max-pooling convolutional neural networks." Neural Networks (IJCNN), The 2012 International Joint Conference on. IEEE, 2012.

https://doi.org/10.1109/IJCNN.2012.6252468

[11] J. W. Shin, "Introducing the Latest Deep Learning Algorithm for Image Recognition", The Journal of The Korean Institute of Communication Sciences 34(7), 6, 25-30, 2017.

[12] H. Chen, et al. "Checkpoint Ensembles: Ensemble Methods from a Single Training Process", arXiv: 1710.03282v1, 2017