Document Classification Using Lightweight Neural Network

Chung-Hsing Chen[1,2], Ko-Wei Huang[1]*

[1] Department of Electrical Engineering, National Kaohsiung University of Science and Technology, Taiwan
[2] Plustek Inc., Kaohsiung City, Taiwan
I110154101@nkust.edu.tw, elone.huang@nkust.edu.tw

Abstract

In recent years, OCR data has been used for learning and analyzing document classification. In addition, some neural networks have used image recognition for training, such as the network published by the ImageNet Large Scale Visual Recognition Challenge for document image training, AlexNet, GoogleNet, and MobileNet. Document image classification is important in data extraction processes and often requires significant computing power. Furthermore, it is difficult to implement image classification using general computers without a graphics processing unit (GPU). Therefore, this study proposes a lightweight neural network application that can perform document image classification on general computers or the Internet of Things (IoT) without a GPU. Plustek Inc. provided 3065 receipts belonging to 58 categories. Three datasets were considered as test samples while the remaining were considered as training samples to train the network to obtain a classifier. After the experiments, the classifier achieved 98.26% accuracy, and only 3 out of 174 samples showed errors.

Keywords: Documents classification, CNN, IoT, Deep learning, Edge computing

1 Introduction

Document classification methods have significantly evolved since 2000. Until 2013, recognition technology was mainly implemented through template matching [1-3] or graph matching [4-7]; however, these methods were mostly used in structured documents that have typesetting and printing with fixed formats, and contain unchanged keywords or fixed patterns on the document, such as the logo of the organization.

Since 2014, deep learning methods have been used to classify document images. Artificial intelligence has developed gradually. For example, the 1993 IBM Deep Blue Computer [8] defeated the then chess Grandmaster Garry Kasparov, and Yann LeCun, also known as the father of artificial intelligence, released LeNet-5 [9]. Although this classic convolutional neural network (CNN) recognized handwritten digits, the application of artificial intelligence stagnated for a long time owing to the insufficient computing performance of computers. Since 2011, after the development of cloud computing and graphics processing unit (GPU) technology, big data analysis through deep learning gained wide attention and was applied to several applications, including document image classification.

Recently, most of the document datasets used the RVL-CDIP [10] or SD2 and SD6 in NIST [11], which are categorized by different natures, such as the content of the RVL-CDIP dataset. RVL-CDIP comprises 16 categories: letter, memo, email, file folder, form, handwritten, invoice, advertisement, budget, news article, presentation, scientific publication, questionnaire, resume, scientific report, and specification, which are different from the problems addressed in this study. This study aims to classify a factory's purchase list according to different suppliers, to recognize the different purchase order formats of different companies.

VLNet [12] was proposed by Chen in 2022 as a basis and header feature of documents as input data for classifier training. This study used VLNet considering the current classification network of document images is too large for devices without GPU. VLNet is a lightweight network that does not rely on GPU operations, and its input makes a set of artificially generated one-dimensional (1D) feature series using a feature sequence with a length of 200 items.

Furthermore, we compared the results of this study with AlexNet and MobileNetV3 experiments to verify the effectiveness of VLNet. Finally, the accuracy of the proposed structure in 174 test samples was evaluated at 98.26%, with only three errors, which was better than 92.53% and 93.68% of AlexNet's two identification intervals, and 97.13% and 95.98% of MobileNetV3's two identification intervals.

2 Related Research

In this section, we discuss the state-of-the-art (SOTA) methods in document image classification, including the use of template matching or document layout to describe the characteristics of files, the use of graphical features to extract document features, and machine learning to classify these features. In the later stages, optical character recognition (OCR) content and machine learning are used to add text rules and classify files, respectively. In the past five years, deep learning has been used to train neural networks to classify files.

Chen *et al.* [13] proposed SHIF to obtain file features that reduce the maximum length and width of the file image to less than 1000 pixels, thereby reducing the number of operations. Furthermore, it uses the scale-invariant feature transform to obtain the descriptors of the feature. Each

*Corresponding Author: Ko-Wei Huang; E-mail: elone.huang@nkust.edu.tw
DOI: 10.53106/160792642023122407012

descriptor contains a 128-dimensional feature vector, which records the strength of the 4×4 bins in an 8-directional plane, and a 4-dimensional vector F, which records the x and y coordinates, scale, and orientation of the descriptor. In addition, they built a k-dimensional tree for these feature vectors in the bin, thereby allowing them to search for the feature vector size in linear time. Using the proposed method, the similarity score for each document category has been calculated by counting the number of nearest neighbor descriptors for the category. Furthermore, we compared the proposed method with architectures previously proposed by Sarkar [14] and Usilin [15]. The accuracy of recognizing a Chinese bank from the database of the Bank of China was found to be 94.96%, whereas that of Sarkar's architecture and Usilin's architecture was only 65.53% and 90.63%, respectively.

Recent research strongly supports the application of deep learning techniques. Krizhevsky et al. [16] proposed the utilization of AlexNet, a well-known deep learning architecture, along with another more streamlined network architecture designed explicitly for document image classification. AlexNet's main architecture comprises five convolutional layers and three fully-connected layers; the last layer outputs the captured features to the first fully-connected layer, which has 4096 dimensions. Another streamlined architecture comprises three convolutional layers and three full layers, wherein the last layer of convolution outputs the captured features to the first fully-connected layer, which has 1000 dimensions. The datasets used in the study were Small CDIP and RVL-CDIP and used two network architectures, different datasets, and different sampling areas for training and experimentation. In the holistic case, the best results are obtained by using the RVL-CDIP dataset and ImageNet init. AlexNet achieved the best accuracy of 89.8%, whereas experiments using Small CDIP datasets obtained 75.6% accuracy. Additionally, when Random Init was used, Small Net and AlexNet achieved 85.1% and 87.8% accuracy in RVL-CDIP, respectively.

To improve the recognition accuracy, Audebert *et al.* [17] proposed a multimodal classifier for hybrid text/image classification using two different networks for feature learning followed by using the features of the two models for classification. Furthermore, the method uses an image CNN network for image feature acquisition. The network architecture is MobileNetV2, and the other network uses OCR text content for encoding, using a 1D convolutional layer to capture features; the data set used in this study is RVL-CDIP and Tobacco3482 [18]. The experimental results showed that RVL-CDIP and Tobacco3482 achieved accuracies of 90.6% and 87.8%, respectively.

Traditional classifier analysis uses a confusion matrix [19] to analyze the classification ability of the classifier. Classification ability refers to the concentration or discreteness of the analysis error. That is, if an error appears in any category, regardless of whether the characteristics of the class are similar, then the classifier cannot correctly distinguish the class; conversely, if the error concentration in certain categories is confusing, it is necessary to rely on other analyses to understand why the classifier concentrates on these categories.

To analyze general CNN networks, Selvaraju *et al.* [20] proposed the Grad-CAM method, which combines feature maps of gradient signals without changing the original network architecture. The primary distinction between this method and the approach introduced by Zhou et al. [21] lies in its capability to visualize the convolutional feature maps without altering the network architecture. Therefore, this method can be applied here to analyze the causes of category confusion. The gradients are set to zero for all classes except the desired class, which is set to 1. The signal is then backpropagated to the rectifier convolutional feature maps of interest, and when it is finally mapped back to the feature map point-by-point, the most interesting features appear in red on the heatmap. The heatmap is then overlapped on the original input image, and the classifier obtains the more interesting features. Recently, several studies have used Grad-CAM [22-24] to understand the features that CNN models focus on.

3 Methodology

For practicality, we used the same dataset to compare classic image classification networks AlexNet and SOTA MobileNetV3 [25]. MobileNetV3 is a lightweight CNN network proposed by Google in 2017 that focuses on mobile or embedded devices, and its biggest innovation is the depthwise separable convolution. We chose MobileNetV3 as a benchmark for comparison because it is similar to VLNet, which was built for the Internet of Things (IoT), thereby allowing us to compare the performance of VLNet and MobileNetV3. AlexNet and MobileNetV3 both perform the classification of images, and hence, can be compared to understand which network is more suitable for small-file classification. This study differs from other studies considering the datasets used, RVL-CDIP and Tabacco3482, are very different file classifications.

3.1 Dataset

The receiving department from Plustek Inc.'s factory can generate over 100 receipts per day, and most receipts record the name of the shipping company, shipping order number, date, material part number, material name, and the quantity of the item. Such data, for companies that do not implement electronic data interchange, can only be manually input into the material management system. Although using automatic identification is used can save time and labor, the form specifications of each company are different. Although the difference is not much, it is best to perform classification first and then identification to identify the form accurately.

As shown in Figure 1, the six samples from 58 categories of purchase orders are documentation of the same nature but from different companies; however, some purchase orders are very similar as shown in Figure 2. This study aims to

classify these groups using a very similar sample. There are 3072 data, and 58 categories have been artificially classified considering the samples are naturally distributed. Because the number of samples in these categories is not balanced, we took three copies of each sample category as test samples, and the remaining were considered as training samples.

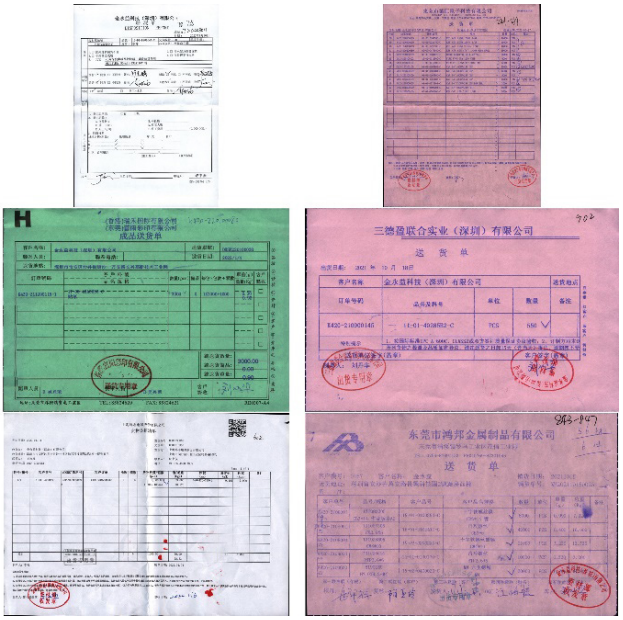

Figure 1. Six samples in 58 categories

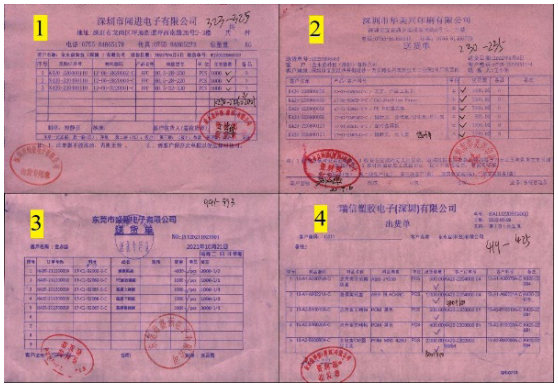

Figure 2. Similar category of dataset

3.2 Region Strategy for Sampling

For rapid classification, only local features have been considered in this study. We assumed that the orientation of the images is correct, considering the rotation of images does not concern this study. Because the general purchase list comprises a meter head and a watch body, the content of the watch body will differ for incoming items. Therefore, we considered a relatively stable table head to capture the features and used this feature series as our training data. The feature regions are determined using Algorithm 1.

Algorithm 1.

$$
Header\ Region = \begin{cases} Region.x = Image.width \times 0.05 \\ Region.y = Image.height \times 0.02 \\ Region.weidth = Image.weidth \times 0.85 \\ Region.Height = Image.height \times 0.15, \\ when\ Image.width > Image.Height \\ \qquad and \\ Region.x = Image.width \times 0.05 \\ Region.y = Image.height \times 0.01 \\ Region.weidth = Image.weidth \times 0.85 \\ Region.Height = Image.height \times 0.08, \\ when\ Image.width \le Image.Height \end{cases}
$$

Because there are two paper formats, portrait and landscape, if the same area algorithm would have been used in the experimental sample, it would have affected the effectiveness of the sampling area. Therefore, as shown in the above formula, the landscape and portrait paper algorithms are slightly different, especially the sampling area of the y-axis.

As shown in Figure 3, the gray block is the block we want to calculate the feature. Therefore, we divide this feature into equal 40 × 5 blocks, and obtain a total of 200 small blocks to calculate the average brightness of this feature. Each feature block will have numbers from 0 to 255, which will be collected into a sequence of features for training.

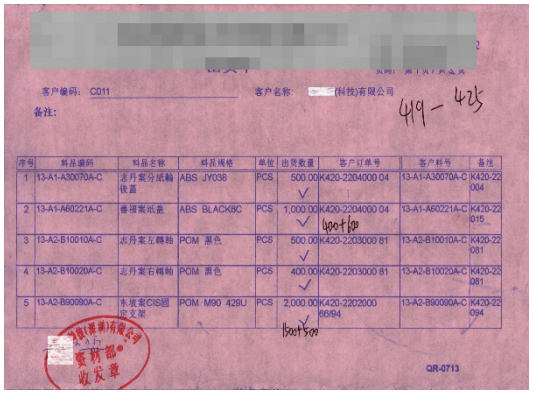

Figure 3. Features map of the header region

The following feature series is obtained after the average brightness of the 200 small blocks: “170, 166, 162, 168, 172, 168, 172, 163, 168, 171, 176, 173, 171, 174, 171, 178, 173, 173, 173, 170, 173, 171, 171, 170, 169, 171, 172, 172, 172, 170, 172, 168, 166, 170, 168, 169, 167, 168, 169, 164, 170, 139, 141, 163, 161, 172, 141, 138, 163, 161, 166, 142, 139, 165, 164, 161, 132, 140, 156, 157, 163, 135, 133, 158, 158, 165, 131, 134, 164, 166, 161, 138, 143, 161, 159, 165, 137, 134, 162, 158, 172, 153, 151, 165, 133, 170, 139, 147, 166, 136, 173, 144, 148, 169, 129, 171, 138, 144, 166, 131, 172, 135, 141, 166, 149, 173, 136, 137, 162, 164, 172, 146, 150, 162, 163, 168, 140, 140, 158, 160, 171,126, 134, 165, 169, 170, 152, 149, 167, 165, 170, 151, 156, 172, 171, 170,

147, 152, 171, 169, 169, 157, 161, 173, 168, 166, 167, 172, 172, 166, 169, 169, 167, 170, 165, 168, 167, 161, 157, 152, 166, 165, 153, 144, 142, 159, 162, 162, 160, 156, 162, 166, 154, 143, 141, 165, 169, 156, 144, 141, 166, 169, 158, 143, 144, 167, 170, 156, 140, 144, 161, 166, 153, 156, 158, 168, 173, 157, 157, and 159." After normalization, this feature sequence becomes a floating-point number between 0 and 1, which will be used by VLNet for learning and obtaining experimental data for this study. The normalization formula is given as:

$$n = \left(\frac{f}{127.5}\right) - 1.0 \ . \tag{1}$$

where n and f represent the normalized and feature values, respectively.

3.3 Lightweight Convolutional Network

To achieve the goal of running even in the IoT, it is necessary to streamline the structure of the network to effectively reduce the number of operations. We adopted the VLNet proposed by Chen *et al.* [12] to the characteristics of document classification. VLNet was published for character recognition, and its input features are 135, including 36 vector features for stoke, 18 edge features for characters, and 81 density features. Its network layer inputs 135 features, after a 1D convolution of mask 1 × 3, a 1 × 133 × 6 feature map, followed by a 1 × 2 max pooling, then a 1 × 66 × 6 feature map, and then a 1 × 64 × 18 feature map after the second 1D convolution, and 52 categories through FC-400-300. Because the input of this study is not a character but a local image of the header region, that is, a 1D feature sequence obtained by sampling, this series is already a sampling result. To reduce the number of layers and acquire more feature data, we deleted the max pooling layer and only connected the two layers of 1D convolution. As a result, the simplified network architecture first enters a series of 200 feature values, the 1D convolution of 1 × 3 obtains the feature map of 1 × 198 × 6, and then the second 1D convolution obtains the feature map of 1 × 196 × 18. Finally, after FC 300|150, 58 categories are output, as shown in Figure 4.

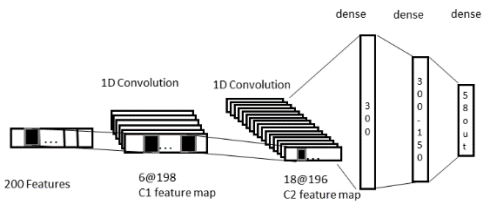

Figure 4. Architecture for lightweight convolutional network

3.4 Experimental Architecture

The performance of this study has been verified by comparing the proposed architecture with the experimental results of AlexNet and MobileNetV3 using the same samples and experimental procedures. This study focuses on the header region, which is in line with the proposed architecture, and the remaining two control networks perform two sets of tests: header and holistic regions. The test is divided into accuracy and elapsed-time tests.

4 Experimental Results

To solve the problem of different receipts of the same file type and to apply it to general IoT devices, our experiment focuses on understanding the correctness of classification as well as emphasizing the computational performance of the proposed method. We compared the accuracy and operation speed of the experimental results of each sampling area of each network.

Our experimental platform uses the Raspberry Pi Foundation's Raspberry Pi Model B with 4 GB of RAM, and the CNN's framework uses PyTorch.

For the test samples in this experiment, three samples were considered for each category, while 58 categories indicated 174 test data. As shown in Figure 5, MobileNetV3 performs better than AlexNet, irrespective of whether it is using the image of the head interval alone or the entire image for training. Nonetheless, both networks performed better than using the head interval alone in all intervals. However, neither MobileNetV3 nor AlexNet obtained results better than VLNet. Regarding the input data, although this study only needed 200 tensors, the number of input parameters of AlexNet and MobileNetV3 was much larger than that in VLNet; consequently, good results can be obtained with a relatively small amount of data. Therefore, we can conclude that VLNet is more suitable for files of the same type with only minor differences. Table 1 presents the detailed experimental results, with different experimental results in different sampling strategies.

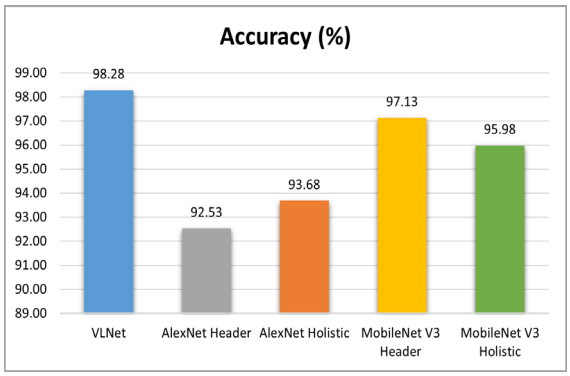

Figure 5. Accuracy results

Table 1. Detailed experimental results

Approach type	Region strategy	Incorrect	Total	Accuracy
VLNet	Header	3	174	98.28
AlexNet	Header	13	174	92.53
AlexNet	Holistic	11	174	93.68
MobileNet V3	Header	5	174	97.13
MobileNet V3	Holistic	7	174	95.98

Figure 6 shows the confusion matrix of the experimental results. The error categories were Class 24 and 45. For Class 24, there were 3 test samples, 1 being wrong and confused with Class 54. For Class 45, there were 3 test samples, 2 being wrong and confused with Class 46.

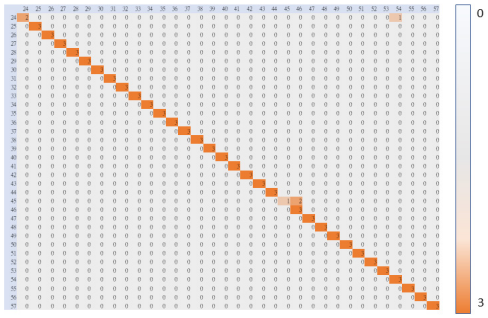

Figure 6. Confusion matrix of experimental results

Figure 7 shows that the header region of the test sample is Class 24; this sample is predicted in Class 54 shown below. The two categories have similarities in the layout.

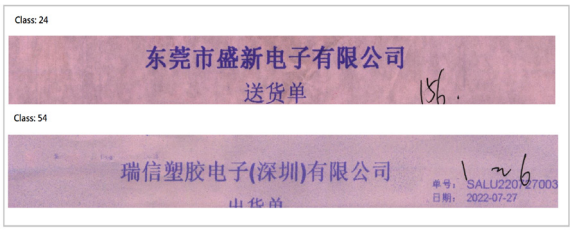

Figure 7. Error sample Class 24 of experimental results

As shown in Figure 8, the two wrong samples are Class 45, which are classified as Class 46. The two categories have similarities in the layout.

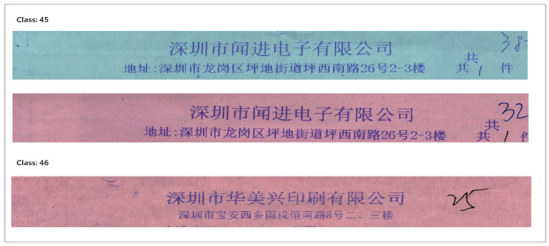

Figure 8. Error sample Class 45 of experimental results

We used Grad-CAM to observe the wrong samples and know that their layout is similar, and used the heatmap in this study to analyze the cause of the error and reveal the interesting features in the CNN.

As shown in Figure 9, the heatmap for class 24 was used to analyze the causes of errors, while Figure 10 represents the heatmap analysis for class 45, the heatmap indicates the primary area of interest in red, the secondary area of interest in yellow, and the area of no interest in blue. This shows that the distribution of red areas is very similar, wherein there are some hot areas around the handwritten digits. Furthermore, there will be some shifts owing to the mapping of the heatmap to the original map, resulting in partial overlap of the handwriting area of the above figure; however, this does not affect our observation.

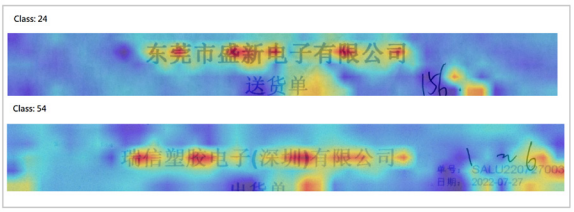

Figure 9. Heatmap for error sample Class 24

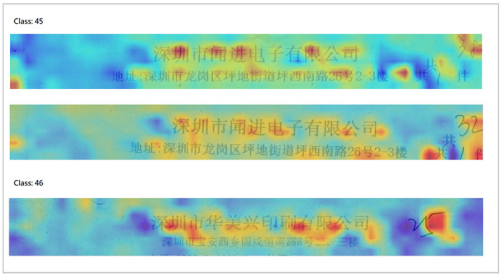

Figure 10. Heatmap for error sample Class 45

As displayed in Table 2, the distribution of hot regions is almost the same in the confused category, which caused confusion in Class 24. We presume that, without OCR, only features should be used to classify. In the above two cases, it is a confusing situation, and not all test samples will be confused in these two classifications; other categories are completely error-free. Nonetheless, the classifier can still accurately classify.

Table 2. Elapsed time

Approach type	Region strategy	Total elapsed time (ms)	Unit elapsed time (ms)
VLNet	Header	1068.22	6.14
AlexNet	Header	292344.55	1680.14
AlexNet	Holistic	317051.77	1822.14
MobileNet V3	Header	38380.32	220.58
MobileNet V3	Holistic	68772.27	395.24

Furthermore, Table 2 shows the experimental results of the elapsed time. The test data was used 10 times for prediction for each approach, and the average time of the 10 runs was considered. The results indicate that MobileNetV3 is several times faster than AlexNet, and meets the original design definition of MobileNetV3. However, there is a significant difference with the proposed method, which is at least 36 times better. Furthermore, considering current computers have a fast central processing unit and GPU, the performance is not significantly affected. However,

performance issues may arise if it is to be implemented on IoT. We used the Raspberry Pi 4 Model B, which took an average of 6.14 ms to classify each document.

5 Conclusions and Future Work

This study solved a practical problem in the classification of files used for the same purpose, but having different formats for subsequent data collection. The results of the experiment verified the original hypothesis: comparing the two latest network architectures, AlexNet and MobileNetV3, with the proposed architecture (VLNet). This study used these two networks as a control group to verify the use of sampling features and a lightweight network to achieve the goal of classification and classify IoT devices. Using the runtime comparison of MobileNetV3, we verified that VLNet is indeed lighter and more suitable for IoT devices than the current streamlined network architectures. The results of the experiment showed that only three errors were obtained in the test set of 174 samples, while AlexNet and MobileNetV3 had 11 and 5 errors, respectively. Therefore, in terms of accuracy, the proposed architecture performed much better. Furthermore, in terms of operating speed, the proposed architecture was 280 times faster than AlexNet and 36 times faster than MobileNetV3.

Because the sample in this study was normally distributed, not every category was balanced, which would have affected the prediction effect of the network. However, to present the actual situation, we did not manually adjust the data set to make it balanced. In the future, we will optimize the network for more samples, and propose this architecture to verify that the deepening of CNNs does not solve the problems encountered in the real world and that the human perspective can be used to summarize artificially made features and input to deep learning, so that the network learns the correct features and removes unwanted interference to improve prediction accuracy.

Acknowledgment

The samples used in this study were provided free of charge by Plustek Inc. We thank the General Manager Lin of Plustek Inc. For their support, and Advanced View Technology Inc. for providing Nvidia GPU for the model training. This study has been successfully implemented, and the results have been introduced into the Plustek DOCaptures cloud recognition platform.

References

[1] Y. Byun, Y. Lee, Form Classification Using DP Matching, *Proceedings of the 2000 ACM Symposium on Applied Computing*, Como, Italy, 2000, pp. 1-4.

[2] H. Peng, F. Long, Z. Chi, W.-C. Siu, Document Image Template Matching Based on Component Block List, *Pattern Recognition Letters*, Vol. 22, No. 9, pp. 1033-1042, July, 2001.

[3] P. Sarkar, Learning Image Anchor Templates for Document Classification and Data Extraction, *2010 20th International Conference on Pattern Recognition*, Istanbul, Turkey, 2010, pp. 3428-3431.

[4] F. Cesarini, M. Lastri, S. Marinai, G. Soda, Encoding of Modified X-Y Trees for Document Classification, *Proceedings of Sixth International Conference on Document Analysis and Recognition*, Seattle, WA, USA, 2001, pp. 1131-1136.

[5] A. D. Bagdanov, M. Worring, Fine-grained Document Genre Classification Using First Order Random Graphs, *Proceedings of Sixth International Conference on Document Analysis and Recognition*, Seattle, WA, USA, 2001, pp. 79-83.

[6] I. Perea, D. López, Syntactic Modeling and Recognition of Document Images, *Structural, Syntactic, and Statistical Pattern Recognition: Joint IAPR International Workshops, SSPR 2004 and SPR 2004*, Lisbon, Portugal, 2004, pp. 416-424.

[7] J. Van Beusekom, D. Keysers, F. Shafait, T. M. Breuel, Distance Measures for Layout-Based Document Image Retrieval, *Second International Conference on Document Image Analysis for Libraries (DIAL'06)*, Lyon, France, 2006, pp. 1-11.

[8] F.-H. Hsu, IBM's Deep Blue Chess Grandmaster Chips, *IEEE Micro*, Vol. 19 , No. 2, pp. 70-81, March-April, 1999.

[9] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-Based Learning Applied to Document Recognition, *Proceedings of the IEEE*, Vol. 86, No. 11, pp. 2278-2324, November, 1998.

[10] A. W. Harley, A. Ufkes, K. G. Derpanis, Evaluation of Deep Convolutional Nets for Document Image Classification and Retrieval, *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, Tunis, Tunisia, 2015, pp. 991-995.

[11] M. D. Garris, NIST Scoring Package Certification Procedures, No. 5173, April, 1993.

[12] C.-H. Chen, K.-W. Huang, English Characters Recognition by Stroke Features and Lightweight Artificial Intelligence, Available at SSRN 4264291, November, 2022.

[13] S. Chen, Y. He, J. Sun, S. Naoi, Structured Document Classification by Matching Local Salient Features, *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, Tsukuba, Japan, 2012, pp. 653-656.

[14] P. Sarkar, Image classification: Classifying Distributions of Visual Features, *18th International Conference on Pattern Recognition (ICPR'06)*, Hong Kong, China, 2006, pp. 472-475.

[15] S. Usilin, D. Nikolaev, V. Postnikov, G. Schaefer, Visual Appearance Based Document Image Classification, *2010 IEEE International Conference on Image Processing*, Hong Kong, China, 2010, pp. 2133-2136.

[16] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet Classification with Deep Convolutional Neural Networks, *Communications of the ACM*, Vol. 60, No. 6, pp. 84-90, June, 2017.

[17] N. Audebert, C. Herold, K. Slimani, C. Vidal, Multimodal Deep Networks for Text and Image-Based

Document Classification, *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Wurzburg, Germany, 2020, pp. 427-443.

[18] M. Z. Afzal, A. Kölsch, S. Ahmed, M. Liwicki, Cutting the Error by Half: Investigation of Very Deep CNN and Advanced Training Strategies for Document Image Classification, *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, Kyoto, Japan, 2017, pp. 883-888.

[19] J. T. Townsend, Theoretical Analysis of an Alphabetic Confusion Matrix, *Perception & Psychophysics*, Vol. 9, No. 1, pp. 40-50, January, 1971.

[20] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, D. Batra, *Grad-CAM: Why Did You Say That?*, arXiv Preprint arXiv: 1611.07450, November, 2016. https://arxiv.org/abs/1611.07450

[21] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Learning Deep Features for Discriminative Localization, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 2016, pp. 2921-2929.

[22] L. Chen, J. Chen, H. Hajimirsadeghi, G. Mori, Adapting Grad-CAM for Embedding Networks, *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, Snowmass Village, CO, USA, 2020, pp. 2794-2803.

[23] R. Fu, Q. Hu, X. Dong, Y. Guo, Y. Gao, B. Li, Axiom-Based Grad-CAM: Towards Accurate Visualization and Explanation of CNNs, arXiv Preprint arXiv: 2008.02312, August, 2020. https://arxiv.org/abs/2008.02312

[24] H. Panwar, P. K. Gupta, M. K. Siddiqui, R. Morales-Menendez, P. Bhardwaj, V. Singh, A Deep Learning and Grad-CAM Based Color Visualization Approach for Fast Detection of COVID-19 Cases Using Chest X-ray and CT-Scan Images, *Chaos, Solitons & Fractals*, Vol. 140, Article No. 110190, November, 2020.

[25] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, H. Adam, Searching for mobilenetv3, *Proceedings of the IEEE/CVF international conference on computer vision*, Seoul, Korea (South), 2019, pp. 1314-1324.

Biographies

Chung-Hsing Chen received his master's degree at the Department of Information Management, National Sun Yat-Sen University, in 2006. Currently, he is the Director of Research and Development Department of Plustek Inc. His current research interests mainly include, network applications, embedded systems and AI image recognition.

Ko-Wei Huang received his PhD from the Institute of Computer and Communication Engineering, Department of Electrical Engineering, National Cheng Kung University, Tainan, Taiwan, in 2015. He is currently an Associate Professor at the Department of Electrical Engineering, National Kaohsiung University of Science and Technology, Taiwan. His current research interests mainly include data mining, deep learning, evolutionary computing, and medical image processing.