applied sciences

MDPI

Article

Large-Scale Printed Chinese Character Recognition for ID Cards Using Deep Learning and Few Samples Transfer Learning

Yi-Quan Li [1,2], Hao-Sen Chang [1] and Daw-Tung Lin [1,*]

[1] Department of Computer Science and Information Engineering, National Taipei University, 151, University Rd., San-Shia, New Taipei City 237303, Taiwan; jerry_li@orbit.com.tw (Y.-Q.L.); box15975316@gmail.com (H.-S.C.)
[2] Orbit Technology Inc., 5F, No. 126, Minzhu West Road, Datong District, Taipei City 10342, Taiwan
* Correspondence: dalton@mail.ntpu.edu.tw

Abstract: In the field of computer vision, large-scale image classification tasks are both important and highly challenging. With the ongoing advances in deep learning and optical character recognition (OCR) technologies, neural networks designed to perform large-scale classification play an essential role in facilitating OCR systems. In this study, we developed an automatic OCR system designed to identify up to 13,070 large-scale printed Chinese characters by using deep learning neural networks and fine-tuning techniques. The proposed framework comprises four components, including training dataset synthesis and background simulation, image preprocessing and data augmentation, the process of training the model, and transfer learning. The training data synthesis procedure is composed of a character font generation step and a background simulation process. Three background models are proposed to simulate the factors of the background noise patterns on ID cards. To expand the diversity of the synthesized training dataset, rotation and zooming data augmentation are applied. A massive dataset comprising more than 19.6 million images was thus created to accommodate the variations in the input images and improve the learning capacity of the CNN model. Subsequently, we modified the GoogLeNet neural architecture by replacing the fully connected layer with a global average pooling layer to avoid overfitting caused by a massive amount of training data. Consequently, the number of model parameters was reduced. Finally, we employed the transfer learning technique to further refine the CNN model using a small number of real data samples. Experimental results show that the overall recognition performance of the proposed approach is significantly better than that of prior methods and thus demonstrate the effectiveness of proposed framework, which exhibited a recognition accuracy as high as 99.39% on the constructed real ID card dataset.

Keywords: large-scale image classification; printed Chinese character recognition; data synthesis; GoogLeNet-GAP; transfer learning

check for updates

Citation: Li, Y.-Q.; Chang, H.-S.; Lin, D.-T. Large-Scale Printed Chinese Character Recognition for ID Cards Using Deep Learning and Few Samples Transfer Learning. *Appl. Sci.* **2022**, *12*, 907. https://doi.org/10.3390/app12020907

Academic Editor: Yuan-Kai Wang

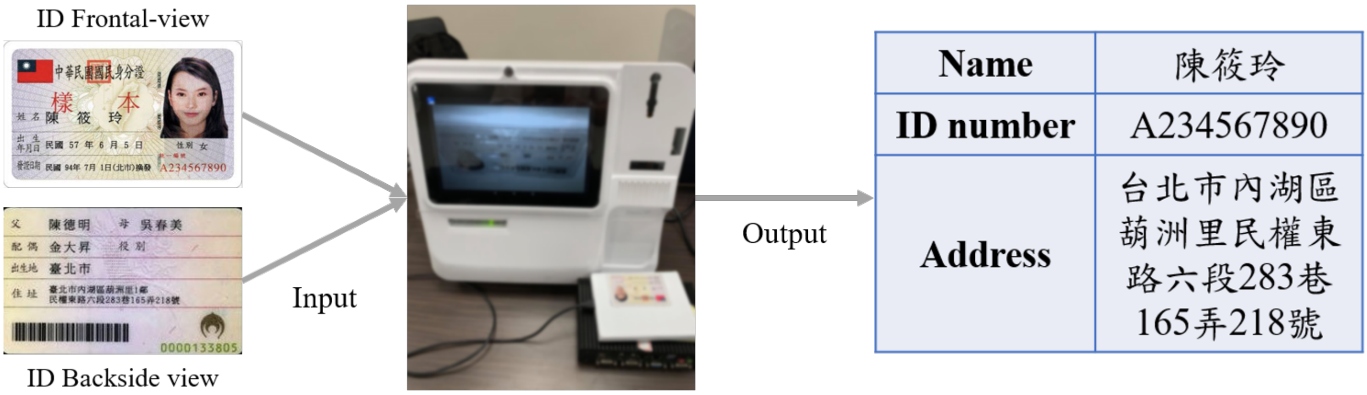
Received: 20 December 2021
Accepted: 14 January 2022
Published: 17 January 2022

1. Introduction

Image classification has always been one of the prominent topics in deep learning, and Chinese character recognition is one application of it. Traditionally, optical character recognition (OCR) has been used for text recognition and it has achieved good results. Large-scale image classification is an important and challenging task in the field of computer vision, which plays an essential role in facilitating OCR methods. For example, the number of classes for a Chinese OCR system could be as high as 13,070. When performing large-scale classification, the amount of data in each category is considered the most important factor. By contrast, if a classifier is divided into an excessive number of characters, the accuracy decreases with an increase in the numbers of characters.

ID card information verification is widely performed for multiple purposes on various occasions, such as for opening bank accounts or making deposits, hotel check-in, clinic registration, identity verification at facility entrances, and pick-up services for purchased items. The development of a system designed to automatically identify personal data on ID cards is expected to provide considerable convenience for both customers and service providers. It also saves human resources and reduces the possibility of errors. It not only saves time but also reduces physical contact, especially when infectious diseases are prevailing, making it especially important and safe. Aprillian et al. [1] developed an OCR system to identify new ID cards issued in Indonesia. Similarly, Purba et al. [2] used Maximally Stable Extremal Regions (MSER) to detect preprocessed Indonesian ID card images, and found the areas where the text was located. Satyawan et al. [3] applied a series of image processing techniques, such as image binarization, Sobel edge detection, and morphology to mark the areas of text on citizen ID cards. Then, Google Tesseract was used as a primary framework to preform character recognition. Their approach correctly identified citizen ID cards at a rate of over 90%. Tavakolian et al. [4] designed a network model called an efficient and accurate scene text detector, which was able to accurately extract the text area. Compared with the MERS-based algorithm, it was more adaptable to natural noise and faster.

On arriving at a hotel, conventionally, one must check-in via the reception staff. The traditional method involves manual data entry on a workstation computer through the manual confirmation of identity documents. Although a bar code is included in contemporary ID cards that records personal information, only government agencies or specific institutions can access and use it owing to privacy issues; ordinary hotels cannot use this feature. Human error is typical under such settings. For example, during peak tourist season, the influx of a large number of customers may easily cause the reception staff to panic or otherwise perform imperfectly, leading to data entry errors. Moreover, customers' privacy may be easily violated and their personal information may be exploited by the malicious actions of staff at the reception desk itself. In this study, we aim to establish a self-service check-in system, as shown in Figure 1, utilizing the proposed large-scale printed Chinese OCR with deep learning and transfer learning. When a user presents their ID card to the camera device, the system can automatically recognize their personal information and complete the follow-up check-in procedures, which not only solves the manual error problem but also avoids the possibility of malicious actions.

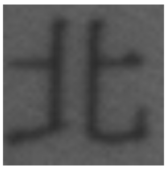


Name	陳筱玲
ID number	A234567890
Address	台北市內湖區葫洲里民權東路六段283巷165弄218號

Figure 1. Prototype of the self-service check-in system.

Despite the fact that handwritten text recognition algorithms have been established, further research on text recognition on ID cards is necessary to account for differences in background and lighting. Background noise tends to cause issues with character segmentation and recognition. Furthermore, after a long period of use, ID cards can develop scratches and damage, and the lamination layer of their cover can yellow with age. The intertwining of these factors makes identification more challenging. The existence of numerous Chinese characters also poses an obstacle to Chinese character recognition methods. Chinese characters, in contrast to numerals or English alphabets, number in the tens of thousands, and even the most regularly used Chinese characters are divided into almost 4000 types. Furthermore, most existing studies have focused on handwritten text, with only a few works attempting to classify or identify printed text. The collection of handwritten text, which is usually generated in a laboratory or particular area, was performed intentionally in the present work. The text image was clear, and the image quality improved. Table 1 compares the characteristics of the handwritten character images and our collected ID card character images. We used dataset synthesis, background simulation, and transfer learning to increase the flexibility and performance of the proposed ID card text recognition technique.

Table 1. Comparison between handwritten text and our ID card text.

	Handwritten Text	**Our Input Text**
Sample Image "North"		
Complexity	High	High
Background	Clean	**Noisy**
Image Size	Specify	**Small**

These challenges motivated us to study the problem of a large-scale classification model for ID card character recognition. In this study, we developed an automatic OCR system to identify up to 13,070 large-scale printed Chinese characters using deep learning neural networks and fine-tuning techniques. As shown in Figure 2, our framework consists of four components. (1) Training dataset synthesis is used to generate a synthesized training set of character images with different Chinese character fonts and simulated ID card backgrounds. (2) Data augmentation is performed on the synthesized training dataset by applying rotation and zooming to increase the diversity of the images contained. (3) The synthesized training dataset is input to the modified GoogLeNet Inception network (GoogLeNet-GAP) [5] for training. (4) Finally, we collected several samples of real Chinese characters on ID cards and performed data augmentation and balancing processing for further transfer learning; finally, the recognition results were obtained as the output.

The main contributions of this work are summarized as follows.

1. We developed a large-scale OCR system to identify printed Chinese characters using a deep learning neural network.
2. We propose a new method to generate printed Chinese characters that simulates background noise patterns found on ID cards. To handle the range of input image changes and increase the recognition capability of a convolutional neural network (CNN) model, a massive dataset of more than 19.6 million photos was developed.
3. We improved the recognition performance of the GoogLeNet-GAP model by incorporating transfer learning.
4. The experimental results demonstrate the effectiveness of the proposed framework; the accuracy of the large-scale 13,070-character recognition system was as high as 99.39%, as evaluated on our dataset of images of real ID cards.

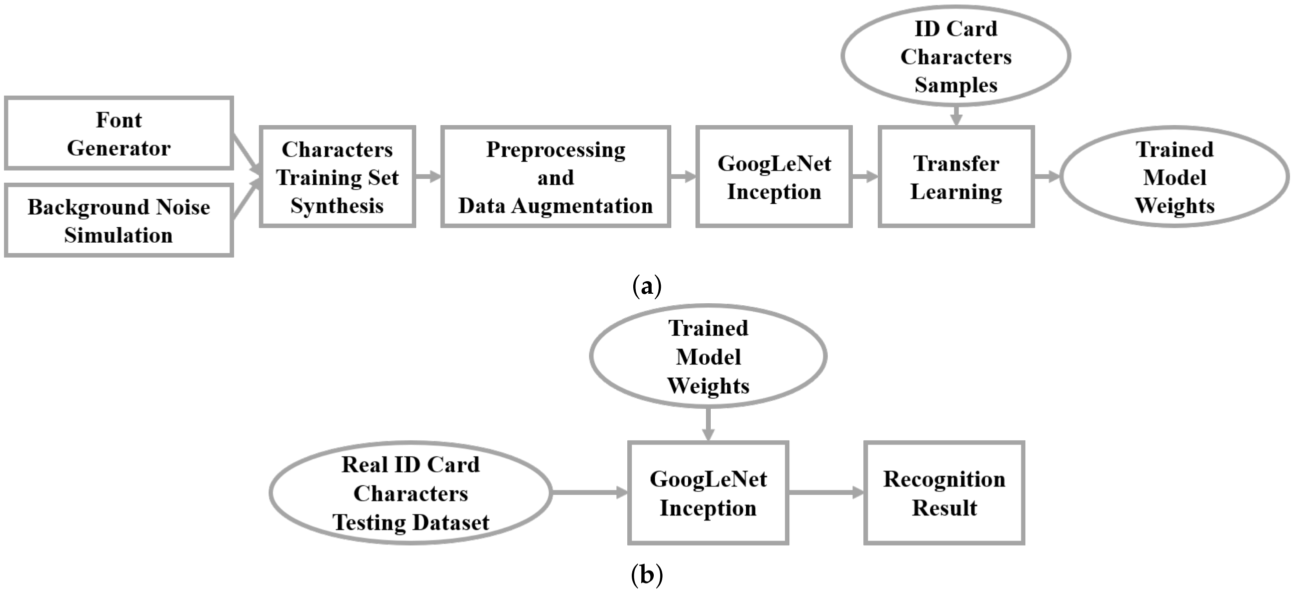

Font Generator
Background Noise Simulation
Characters Training Set Synthesis
Preprocessing and Data Augmentation
GoogLeNet Inception
ID Card Characters Samples
Transfer Learning
Trained Model Weights
(a)
Trained Model Weights
Real ID Card Characters Testing Dataset
GoogLeNet Inception
Recognition Result
(b)

Figure 2. Flowchart of the proposed large-scale Chinese character recognition system: (**a**) Training phase and (**b**) recognition phase.

2. Related Work

2.1. Deep Learning Chinese Character Recognition

Optical character recognition (OCR) is a mainstream text recognition technology. Through the analysis and comparison of a pre-built database, various texts can be quickly recognized. With the tremendous advancement of deep learning-based computer vision technology in recent years, the convolutional neural network (CNN) approaches have been greatly applied to OCR. Xu et al. [6] proposed an end-to-end subtitle recognition system. After inputting a video with subtitles, the subtitle area is marked, and sliding windows are used to cut the characters one by one at regular intervals and recognize them. Although an end-to-end system is adopted, the sliding window can easily cut out too many repetitive characters and cause failure. Zhong et al. [7] proposed an HCCR-GoogLeNet, which reduces the parameters of the original GoogLeNet [5], and adds the traditional feature extraction method HoG, and obtain gradient feature maps to enhance the performance of CNN. The recognition rate of the model evaluated with the Chinese Academy of Sciences handwriting databases (CASIA-HWDB) dataset [8] attained 96.74%. Lin et al. [9] proposed new architecture to avoid the overfitting problem under a multi-classification problem. By using global average pooling (GAP) to replace the original fully connected (FC) layer, it can reduce the number of parameters while maintaining the original effect. Li et al. [10] resolved the disadvantage of the traditional CNN, which would require huge computation resources, and improved the recognition effect by increasing the output layer of the intermediate layer. However, GAP will cause a significant decrease in accuracy, hence they added trainable weights to GAP to solve this problem which is called global weighted average pooling (GWAP). The model achieved an accuracy of 97.1% for the CASIA-HWDB1.1 [8] dataset. Xiao et al. [11] proposed a new technology, called adaptive drop weight (ADW), which can effectively reduce the number of CNN parameters, and proposed global supervised low-rank expansion (GSLRE) to accelerate the entire CNN model. Compared to other baselines of handwritten Chinese character recognition (HCCR) models, the model reduces the amount of calculation by nine times and compresses the parameter amount to 1/18, but the accuracy rate is only reduced by 0.21%. With its extremely low parameter amount, it can be deployed on a mobile device to achieve rapid recognition. Melnyk et al. [12] proposed new architecture in 2019 to improve the classification accuracy of the output layer by modifying GWAP, and extracting the class activation map (CAM) to perform character recognition. This model was evaluated with the CASIA-HWDB1.1 dataset [8], and the recognition accuracy rate reached 97.61%, breaking the record of the year. Su et al. [13] developed a device that could be worn on the hand. This model can accurately identify 6000 Chinese characters and achieves an accuracy of 96.75% in the Chinese FingerReader testing dataset. However, its research also shows that the overall recognition rate will be affected if it encounters serious background interference. Yao et al. [14] suggested a new text-classification approach based on graph neural networks, which learned word and document embedding by modeling an entire corpus as a heterogeneous graph. The results showed that their proposed approach exhibited better performance compared to state-of-the-art text-classification algorithms. Using text sequence relations in sentences, natural language processing technologies can improve the recognition rate of OCR systems. Liu et al. [15] created a CNN model using a temporal classification loss function. They used dropout after each max-pooling layer to reduce overfitting. They achieved a character error rate of 6.81% on the ICDAR 2013 competition dataset. By adding DenseNet, Wang et al. [16] upgraded a convolutional recurrent neural network and achieved an average character recognition accuracy of 98.57%. A data synthesizer based on conditional adversarial generative networks is also being developed to create synthetic ID card text line graphics. Du et al. [17,18] created PaddleOCR, a lightweight OCR system, and applied several techniques to increase their model's performance, including a light backbone, cosine learning rate decay, learning rate warm-up, pruning, data augmentation, and quantization. EasyOCR [19] is a free and open-source scene text OCR system based on the convolutional recurrent neural network architecture [20], with ResNet and VGG16 models, long short-term memory encoding, and a connectionist temporal classification [21] decoding process.

2.2. Handwritten Chinese Character Dataset

Recent Chinese character recognition researches are mostly evaluated using the Chinese Academy of Sciences handwriting databases (CASIA-HWDB) dataset [8]. CASIA-HWDB [8] was released in the ICDAR 2013 offline HCCR competition, which contains three subsets, CASIA-HWDB1.0, 1.1, and 1.2. Generally subsets 1.0, and 1.2 are used as the training set, and subset 1.1 is used as the test set, which contains 3755 different handwritten Chinese characters. However, these datasets are simplified Chinese and not traditional Chinese. Traditional Chinese and simplified Chinese are quite different in structure. The simplified Chinese omits complicated strokes for the convenience of writing. Therefore, the recognition of simplified Chinese is generally easier than that of traditional Chinese. There are not many traditional Chinese datasets. Among them, Chinese MNIST [22] is the largest handwritten traditional Chinese dataset, which contains 13,065 traditional Chinese characters. Although the number of characters is much larger than that of CASIA-HWDB [8], the amount of samples of each character is far behind. If CASIA-HWDB is used for a large-scale classification training, it is very possible to cause underfitting because of insufficient data. Yue et al. [23] established a database CASIA-AHCDB of ancient characters. It was collected from 11,937 pages of ancient Chinese manuscripts. There are 10,350 characters of different handwritten Chinese. They are mainly divided into two large datasets, each containing three parts of the data, which can be used for different purposes. However, the number of characters differs. If they are used as the training data for a large classifier, the convergence time will be too long or even unable to converge.

2.3. Large-Scale Classification

When performing large-scale classification, the most important factor is the amount of data in each category. To train a high-precision neural network, a large amount of data is necessary to assist, and it is known as data hungry. If the data are insufficient, the model will not be able to converge, or the amount of data in each category will be different, which will easily bias the model to the side with more data. Therefore, in the research conducted so far, there are few relevant interpretations for large-scale classification. Once the classifier is divided into too many characters, the accuracy decreases as the number of characters increases. Although the impact can be reduced by data expansion, it still cannot reach a practical level. The traditional neural network model uses the fully connected (FC) architecture in the final output layer to classify the features extracted in the first half of the CNN. However, because of its many parameters, it not only increases the burden on calculations but also easily causes overfitting problems. Therefore, partial connections are often used to reduce the probability of overfitting. If it is used for Chinese character classification, the impact will be even more serious.

Zhong et al. [24] proposed a multi-pooling method and nonlinear data transformation to improve the effect of large-scale printed Chinese character recognition. Experimental results show that the proposed model achieves good results for 3755 printed Chinese characters. Li et al. [10], Melnyk et al. [12], and Qiu [25] demonstrated through experimental results that GWAP with adaptable parameters can extract regional features better than traditional fully connected layers to improve the classification effect and reduce the burden on calculations. Zhang et al. [26] proposed a label-mapping (LM) strategy. By dividing a huge category into several sub-characters and then predicting each sub-category, the results showed that LM can effectively improve the accuracy of large-scale classification on the CJK characters. Zhang et al. [27,28] proposed the radical analysis network (RAN), which analyzes the two-dimensional spatial structure of Chinese characters and disassembles them into several parts divided by the radicals. Treating characters as a combination of multiple radicals instead of a single character reduces the number of characters that must be recognized. Through the combination of different radicals to identify characters that have never been encountered, Wu et al. [29] proposed the joint spatial and radical analysis network (JSRAN), whose architecture can effectively resolve the problems of huge characters and limited data when traditional Chinese character recognition is executed. To improve remote-sensing imagery classification, Hong et al. [30] blended a general multimodal deep learning model with five fusion architectures.

2.4. Transfer Learning

Transfer learning uses a pre-trained model, and then adopts a small amount of new data as input and retrains the model. Consequently, the pre-trained model learned to digest new information and infer what the old data do not. After several training cycles, the feature extraction method can effectively learn the characteristics of the new data in a short time, while retaining the feature extracted by the old data, thus identifying the new data. Qiao et al. [31] proposed a new Siamese network architecture. Traditionally, to learn new characters from existing models, they must use sufficient characters of data for training however, they use a small amount of data for pre-training and learn new category characteristics by predicting the parameters of the activation function efficiently. Ao et al. [32] used hybrid models by combining RNN and CNN, when recognizing new handwritten characters, the characteristics of the word can be learned through the printed character and do not need the handwritten samples. Using printed characters as a prototype of handwritten characters and training the entire network, a small number of handwritten images are used for fine-tuning. Experimental results show that the proposed method can effectively use printed characters to learn the characteristics of handwritten characters. Tang et al. [33] utilized a large number of printed Chinese characters for model pre-training. After the training, its parameters were initialized as the parameters of the new model and fine-tuned with a small amount of labeled ancient texts and handwritten Chinese characters. The feature extractors and classifiers originally used in printed characters were transferred and adapted to the new data. Tang et al. [34] proposed a semi-supervised transfer learning (STL), which integrates the multi-kernel maximum mean discrepancy (MK-MMD) loss function into the traditional transfer learning model, thereby narrowing the gap between the source and target domains. They first used a large amount of labeled data for CNN training, and then used a small amount of target domain data for fine-tuning. Finally, a large number of unlabeled target domain data (with MK-MMD loss) and a limited target domain of labeled data (with softmax loss) are used to train the entire model simultaneously. It has been tested on several well-known CNN networks, including AlexNet [35], GoogLeNet [5], and ResNet [36]. The experimental results show that the proposed method effectively enhances the accuracy rate of Chinese character recognition.

3. Methodology

The main objective of this study was the construction of an automatic OCR system designed to identify up to 13,070 large-scale printed Chinese characters using deep learning neural networks and transfer learning techniques. As shown in Figure 2a,b, the overall procedure of the proposed framework is divided into two phases: Training and recognition, respectively. The main steps of the training phase are as follows. (1) Using several Chinese character fonts and simulated ID card backgrounds, a synthesized training set of character images was created. (2) Data augmentation was performed on the synthesized training dataset by rotating and zooming to improve diversity. (3) The proposed GoogLeNet-GAP model was trained using the synthesized training dataset. Subsequently, (4) several samples of the Chinese characters were collected from real ID cards, and data augmentation and balancing processing were performed for further transfer learning. Once the network converged, the weights of the trained model were obtained. During the recognition phase, the trained model weights were input into the GoogLeNet-GAP model. Real-life character samples from the ID cards were then fed into the model as input, and the results were displayed. The steps taken are detailed in the subsections below.

3.1. Training Dataset Synthesis

The deep learning mechanism relies on a large amount of training data. However, owing to the privacy issues associated with collecting personal ID card images, the availability of real data remains limited. Hence, the construction of large-scale synthetic data for neural network training was necessary in this study. Furthermore, the appearance of text on ID cards may vary owing to different backgrounds and lighting. Therefore, we created the synthetic Chinese characters training dataset through the following steps. First, (1) generate a font image with a white background. Then, (2) produce a simulated background, and (3) combine the font and background images. The detailed data synthesis process is presented below.

Font Generation. First, we used the Pillow [37] package in the Python programming langue to create a 100 × 100-pixel white image as the background and then pasted the character text on the background to generate a character sample image. Big5 is a common Chinese character encoding method used for traditional Chinese characters, which contains a large set of 13,060 characters used in daily life. This study uses it as the classification target and adds the digits 0–9 that often appear on the ID cards. There were a total of 13,070 characters. Since the most commonly used Chinese fonts are Microsoft JhengHei and MingLiU, we chose these two fonts and added their boldface versions to increase the diversity of the dataset, with a total of four different fonts for each character. In addition, the Times News Roman font was adopted for the digits 0 to 9. Therefore, a total of 52,290 character images were generated.

Background Simulation. Second, to simulate the factors of the background stripes patterns on ID cards, we added noise and merged them into the aforementioned generated character images. In this study, we introduce three background simulation methods, including a random gray-level background, a random noise background, and a patch stitching background, as described below. The three proposed background simulation methods can effectively convey the background pattern of real ID cards and facilitate the learning capability of the CNN model to achieve better OCR performance.

(1) To simulate various environmental lighting changes, we overlaid the generated character images with a different gray-level background, as depicted in Figure 3. The gray level was randomly selected. Additionally, the resultant character image was blurred with a Gaussian filter to smooth the appearance of the generated character image.

(2) The second type of background was random noise. First, a pure gray-level background was generated. Next, a positive or negative random number was added to the gray-level background. Then, the character image was merged with the noisy background, as shown in Figure 4. Finally, Gaussian blur was applied to smoothen the appearance.

(3) To improve the instability caused by random noise, we further propose a patch stitching method to simulate the ID card background. First, we randomly selected 50 patches of 2 × 2 images from the surrounding areas of a real ID card image, as shown in the red boxes in Figure 5a. Next, these patches were stitched in order from left to right and from top to bottom into a 100 × 100-pixel background image, as shown in Figure 5b. After obtaining the stitching background, it was combined with the character image, and Gaussian blurring was performed. Figure 6 illustrates the process.

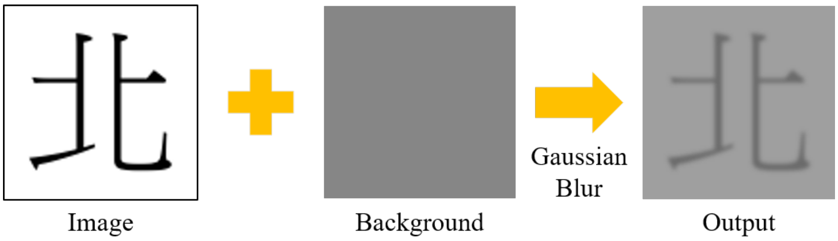


Figure 3. Background simulation method 1: Character "North" image with a gray-level background.

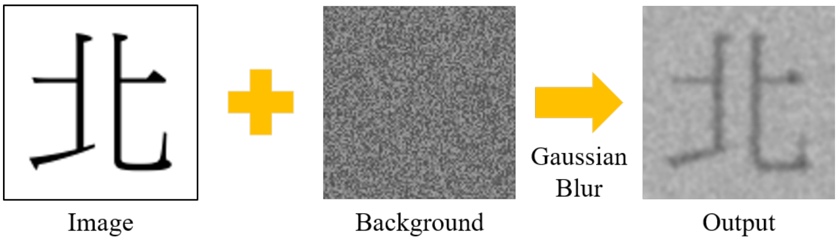


Figure 4. Background simulation method 2: Character "North" image with random noise background.

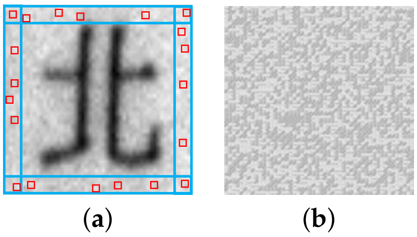


Figure 5. Stitching background method: (**a**) Random selection of patches and (**b**) stitching background result.

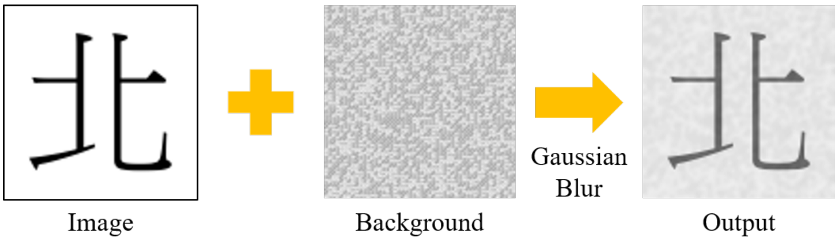


Figure 6. Background simulation method 3: Character "North" image with patch-stitching background.

3.2. Image Preprocessing and Data Augmentation

Grayscale Normalization. The problem of image brightness range variation is often encountered in the OCR task. To enhance the images, preprocessing of the brightness normalization techniques was applied. Based on the preliminary experiments using brightness normalization techniques such as min-max, averaging, and histogram equalization, we adopted the min-max normalization method (Equation (1)) and achieved the best recognition accuracy. Figure 7 shows an example of min-max normalization. Compared to the original image (Figure 7a), the normalized image appears to exhibit better contrast. This enhancement leverages feature extraction for deep learning:

$$X_{minimax} = \frac{X - X_{min}}{X_{max} - X_{min}}. \tag{1}$$

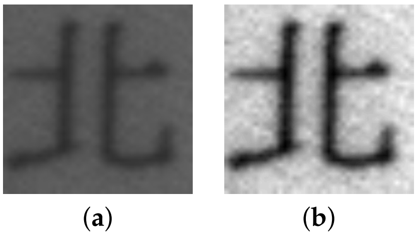

(**a**) (**b**)

Figure 7. Example of normalization preprocessing: (**a**) Original image. (**b**) MiniMax normalization result.

Data Augmentation. To ensure that the proposed model can adapt to a variety of different situations, considering that the ID cards may be aligned in different angles or the ID card images may be captured at slightly different distances from the cameras, the rotation angle and size of the text image are expected to change accordingly. Furthermore, despite the fact that our current YOLOv3-based [38] detection approach achieved 100% correct character cropping, the segmentation mAP has yet to reach 1.0, and the detection bounding box is occasionally slightly rotated or has different sizes. As a result, the rotation and rescaling of data augmentation are required during the training phase. Data augmentation was further applied to the synthesized data generated in Section 3.1, where each image was randomly rotated between 10° and −10°, and enlarged between the ratios of 1.1 to 1.3, so that the diversity of the data increased. Examples of data augmentation images are shown in Figure 8.

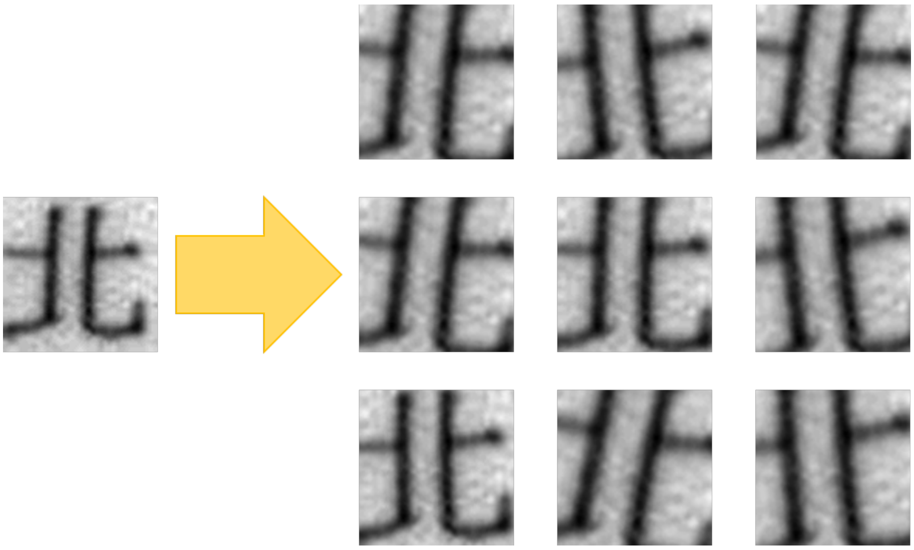


Figure 8. Examples of data augmentation images.

3.3. Baseline Model Pre-Training

Using the different font generation, background simulation, and data augmentation processes illustrated in Sections 3.1 and 3.2, we expanded the dataset to generate a sufficient amount of training data, ultimately using up to more than 19.6 million images to accommodate the diversity of input image variations and strengthen the recognition capability of the CNN model. We were inspired by two studies in the field of handwritten Chinese character recognition (HCCR): HCCR-GoogLeNet [7] and MelnykNet [12]. To conduct a preliminary classification test, we used a portion of the real data gathered, a total of 66 different characteristics. We also used InceptionV4 [39] for comparison. As GoogLeNet and MelnykNet perform better, we chose these two models for investigation.

In this study, we utilized and modified the GoogLeNet [5,7] and MelnykNet [12] models as a baseline and pre-trained them with the abovementioned synthetic Chinese characters dataset. According to Zhong et al.'s research [7], traditional, less demanding CNNs, offered for handwritten Chinese character recognition, were neither deep nor slender enough to extract appropriate features to achieve better classification performance. We did not consider using standard CNNs, such as AlexNet, because GoogLeNet has a deeper architecture and can be developed with fewer parameters to achieve superior performance.

GoogLeNet-GAP model. GoogLeNet [5] was proposed by Google in 2014 and won the ILSVRC competition. By adding different sizes of the kernel to extract the features of images at different scales, they also added 1 × 1 convolution to reduce the amount of calculation, and finally combined these features to form the Inception v1 block [5]. Combining multiple inception blocks can increase the diversity of the feature map and make the model converge more quickly. Since our goal is to classify large-scale categories of Chinese printed characters, to avoid overfitting caused by a large number of characters, we replaced the FC layer with a global average pooling (GAP) layer (as plotted in red-dashed line in Figure 9) to reduce the number of model parameters and improve its classification performance. Figure 9 shows the modified network architecture GoogLeNet-GAP.

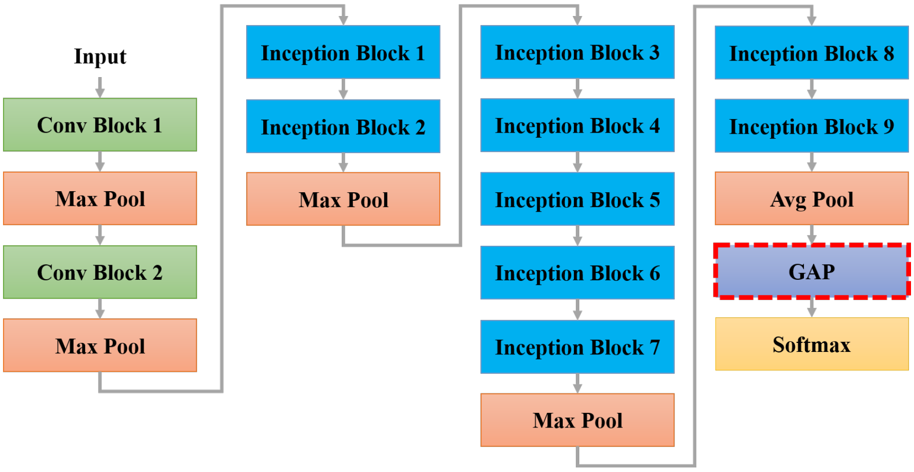


Figure 9. Modified GoogLeNet architecture: GoogLeNet-GAP model.

MelnykNet-Res model. MelnykNet was originally proposed for offline handwritten Chinese character recognition [12]. In the study of Melnyk et al., the handwritten character data were binary images and the background was clear [12] hence, the MelnykNet model only requires a few parameters and can be trained to perform well. However, as our input images are grayscale images with a noisy background, it was necessary to strengthen the feature extraction capabilities of the CNN model to achieve better recognition performance. According to Matthew et al. [40], the feature maps extracted by each convolutional layer differ; the higher the layer, the finer the feature maps, and vice versa. In the original architecture, the contour of the characters were extracted by the previous convolutional layer, so we added a residual block [36] (as plotted in the red-dashed line in Figure 10) to the later layer to continue to extract the detailed features of the character. The modified MelnykNet model MelnykNet-Res is illustrated in Figure 10.

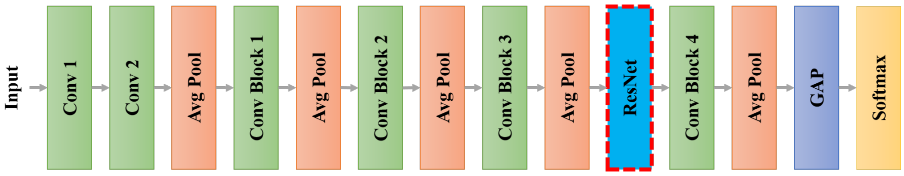


Figure 10. Modified MelnykNet architecture: MelnykNet-Res model.

In this study, we collected a test dataset containing 4944 samples of 294 Chinese characters cropped from real ID card images. Preliminary experiments were conducted using the above-mentioned GoogLeNet-GAP model and MelnykNet-Res model. We performed training using the synthetic training dataset and compared the recognition rates of the two models using a real testing dataset. Table 2 presents the performance of these two models, given the different numbers of training samples augmented for each character. As may be observed from Table 2, the modified GoogLeNet model performed better than the MelnykNet model. The highest recognition rate was 95.71%. Therefore, we chose the modified GoogLeNet-GAP model for further study.

Table 2. Performance comparison of the modified MelnykNet-Res model and GoogleLeNet-GAP model tested on real ID card data.

# of Training Samples per Character	MelnykNet-Res Accuracy	GoogLeNet-GAP Accuracy
300	21.64%	**51.40%**
600	41.75%	**90.62%**
900	53.40%	**95.71%**
1500	68.95%	**95.35%**

3.4. Mixed Data Model and Transfer Learning Model

Although the proposed GoogLeNet-GAP model achieved good recognition accuracy on the real ID card character testing dataset (see Section 3.3), a certain degree of misclassification remained. To improve the performance of our model, we expanded the synthetic training set by including an augmented version of the data from a small part of the real ID card character images. We propose two new models: (1) A mixed data model and (2) a transfer learning model. The mixed data model was obtained by retraining the GoogLeNet-GAP model (see Figure 9) from scratch while mixing synthetic training data and augmentation data of partial samples from real ID card character images. In contrast, the transfer learning model used the pre-trained model (see Section 3.3) as the base model and performed fine-tune training with the augmented data of partial samples from the real ID card character images. The abovementioned two methods were validated through experiments in which both the proposed new models achieve substantial improvements in recognition accuracy. Moreover, the results show that the transfer learning model is more practical for practical applications. The model recognition performance can be gradually improved by periodically fine-tuning with newly-collected real data. Meanwhile, the training speed can dramatically increase because of the very small amount of new data. Section 4 presents the detailed experimental results and reveals the advantages of the transfer learning model.

4. Experimental Results

4.1. Dataset and Implementation Details

As illustrated in Section 3.1, the training dataset was constructed by synthesizing five different fonts of the common Chinese character Big5 code and digits using the Pillow [37] package in Python. A total of 52,290 character images were generated. Furthermore, to accommodate the diversity of input image variations, each of these 52,290 characters was synthesized again with different noisy backgrounds or augmented with the angle rotation and resizing processes described in Section 3.2. Finally, we expanded the dataset to generate a sufficient amount of training data, ultimately including up to more than 19.6 million images. Table 3 lists the detailed content of the dataset mentioned in this study. To the best of our knowledge, no dataset with a sufficient number of Chinese character images to be used for a large-scale classifier has been developed in prior work. The proposed synthesis method can automatically generate character images through programs and can thus create a sufficient amount of training data and include augmentation to increase diversity.

Table 3. Comparison of the common Chinese character datasets and the proposed dataset.

Dataset	# of Characters	# of Images	Type
CASIA-HWDB1.0	3866	1,609,136	Simplified Chinese
CASIA-HWDB1.1	3755	1,121,749	Simplified Chinese
CASIA-HWDB1.2	3319	990,989	Simplified Chinese
Chinese MNIST	13,065	587,925	Traditional Chinese
CASIA-AHCDB	10,350	more than 2.2 million	Traditional Chinese
Ours	**13,070**	**19.6 million**	Traditional Chinese

The synthetic training dataset was divided into three parts; 70%, 10%, and 20% were randomly selected for network training, validation, and testing, respectively. The validation set was used to avoid overfitting during training, while the testing set was used to verify the effects of the trained model. We set the initial learning rate as 0.01 with a 0.1 validation accuracy factor, minimum learning rate as 10^{-5}, dropout rate as 0.4, and mini-batch as 64. We initialized the weights using glorot uniform distribution and employed ReLU activation and a categorical cross-entropy loss function. The proposed model was trained for a maximum of 30 epochs using the Adam optimizer. Our models were implemented in Python 3.7 on the TensorFlow platform under a Windows operating system. The experiments were conducted on workstation computer with an Intel Core i9-9900X CPU at a clock rate of 3.50 GHz with 64 GB of RAM and an NVIDIA GTX 2080 Ti GPU. All of the character identification results in this work were assessed using the same testing dataset, which consisted of 4944 samples of 294 Chinese characters and was derived from 246 genuine ID card photos. Each sample has an image size of roughly 45 × 45 pixels, clipped from ID card photos. Figure 1 shows the apparatus that was used to collect the ID card photos. Aged lamination cover, stain, and scratches are common damages in the testing dataset images, with an 8.5%, 1.6%, and 5.7% distribution, respectively. The remaining samples are undamaged.

4.2. Ablation Study

To verify that our GoogLeNet-GAP network can overcome the overfitting problem and obtain higher recognition accuracy, we compared the performances of the models with the original FC layer and the new GAP layer by providing a different number of training samples per character for the 294 output classes in the GoogLeNet model. Table 4 shows the effect of using the GAP layer as the classification output layer. Our GoogLeNet-GAP network achieved higher accuracy. Moreover, it may be observed that when additional training samples was provided, the network model performed better. Therefore, subsequent experiments will be based on the GoogLeNet-GAP model.

In addition, for the fully connected nature of FC, when the number of character categories increased, the number of FC layer parameters also increased dramatically. Thus, the chances of overfitting increased sharply. When using GAP, because its hidden layer does not have trainable parameters, the parameters of the output layer are only increased on increasing the number of classifications. Therefore, compared with FC, GAP has fewer parameters and is less likely to cause overfitting. Furthermore, the use of fewer parameters can increase the computational speed of the CNN. Table 5 shows the number of parameters used in the FC and GAP for various classification tasks.

The computational cost of the network training phase is determined by the amount of classifications and training samples per character required. It may also be identified in terms of the computing resources and software environment used. Additional experiments on the quantitative assessment of network computational time have been performed. Regarding the GoogLeNet model, Table 5 compares the computational complexity of FC and GAP. As shown, using GAP minimized the time spent on training and recognition.

This research is focused on a real-time application with the goal of deploying the proposed OCR system on a remote server, allowing smartphones to send images to the backend for text recognition. Thus, the character recognition process requires immediate execution. The recognition time for each character was approximately 0.0026 s, as shown in Table 5. Hence, the proposed approach can be used in real-time scenarios.

Table 4. Performance comparison of GoogLeNet models using fully connected (FC) and global average pooling (GAP).

# of Training Samples per Character	FC	GAP
300	41.59%	51.40%
600	87.74%	**90.62%**
900	85.56%	**95.71%**
1500	91.20%	**95.35%**

Table 5. Comparison of parameters size and computational complexity of FC and GAP in GoogLeNet model.

# of Classes (Characters)	GoogLeNet-FC			GoogLeNet-GAP		
	# of Parameters (M)	Training Time (s/per epoch)	Recognition Time (s/per char)	# of Parameters (M)	Training Time (s/per epoch)	Recognition Time (s/per char)
294	11.802	181.8	0.002582	11.716	180	0.002521
1812	16.557	1077.6	0.002603	**13.271**	**1030.6**	**0.002569**
4945	40.941	4824.5	0.002598	**16.483**	**4014**	**0.002588**
13070	195.649	25855	0.003418	**24.811**	**23,443.5**	**0.002622**

According to the Ministry of Education of Taiwan, 4803 standard Chinese characters are frequently used in daily life. There are 1802 characters commonly used in addresses according to the postal system. In addition to our target of the large-scale classification of 13,070 characters, we also conducted experiments on the models of 1812 outputs (1802 characters plus 10 digits) and 4945 outputs (the union of 4803 standard Chinese characters, 1802 characters in postal system, and 10 digits) for the postal system and education system, respectively. Table 6 indicates that the GoogLeNet-GAP maintained classification performance, even when the number of model outputs was expanded from 1812 to 13,070. Furthermore, all the character recognition tasks were performed with an accuracy exceeding 90% when the number of training samples per character increased to 900. The system achieved the best performance on average, given 1500 training samples per character. Nevertheless, the more training samples provided, the more time required to train the model. If the model can be further improved to achieve compatible performance with less training data, it will become more practical for real-time applications.

Table 6. Performance of various large-scale GoogLeNet-GAP models with different numbers of training samples per character.

# of Training Samples per Character	294 Classes (Characters)	1812 Classes (Characters)	4945 Classes (Characters)	13,070 Classes (Characters)
300	51.40%	77.91%	76.52%	66.51%
600	90.62%	88.21%	89.99%	83.72%
900	95.71%	91.91%	89.42%	88.51%
1500	95.35%	95.28%	90.23%	92.17%

4.3. Mixed Data Model and Transfer Learning Model Results

4.3.1. Mixed Data Model

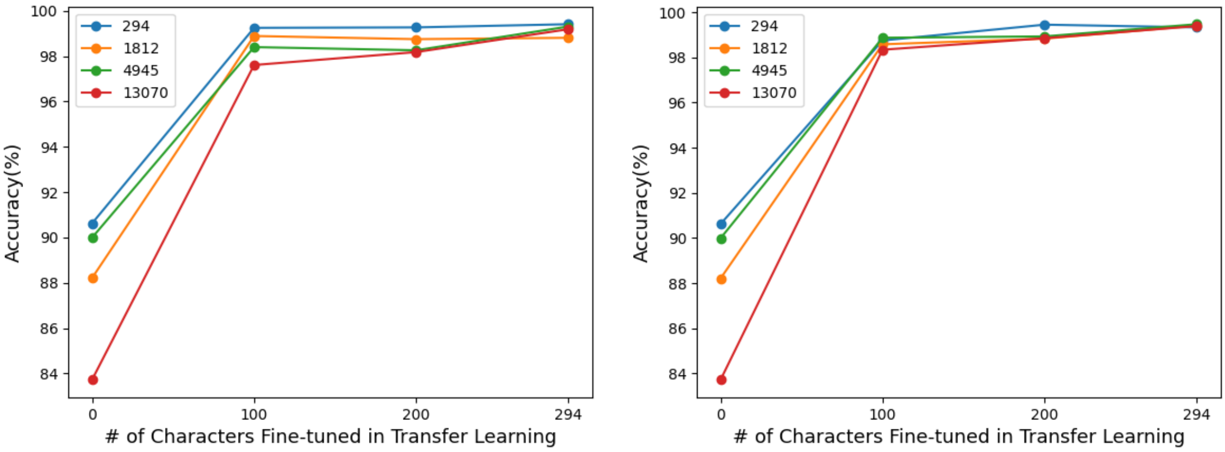
The mixed data model was obtained by retraining the GoogLeNet-GAP model from scratch. The training set was composed of the augmented versions of partial real data and partial synthetic training data created previously (see Sections 3.1 and 3.2). We randomly selected 66 of the 294 characters from the real data. Then, rotation and resizing augmentation are applied to the original 235 images of these 66 characters and then expanded 20% of the number of training samples of each character. For instance, if the number of training samples per character was 600, then 120 samples were obtained from real data augmentation and 480 samples were taken from the synthetic dataset of the corresponding character. Finally, the model was re-trained. Table 7 shows that the accuracy increased significantly, not only for the effect of the original 66 characters but also for the new characters. Although the results obtained by this method were good, there were still certain drawbacks. Once new real data are available, the model must be re-trained after the new data are mixed with the original data. This causes a substantial increase in training time, and thus, this function cannot be used for real-time recognition services.

Table 7. Performance comparison of the original model and the mixed data model with 600 and 900 training samples per character.

# of Classes	600 Training Samples per Character		900 Training Samples per Character	
(Characters)	Original Model	Mixed Data Model	Original Model	Mixed Data Model
294	90.62%	**96.84%**	95.71%	**98.51%**
1812	88.21%	**96.22%**	91.91%	**95.03%**
4945	89.99%	**93.54%**	89.42%	**96.16%**
13,070	83.72%	**91.95%**	88.51%	**93.93%**

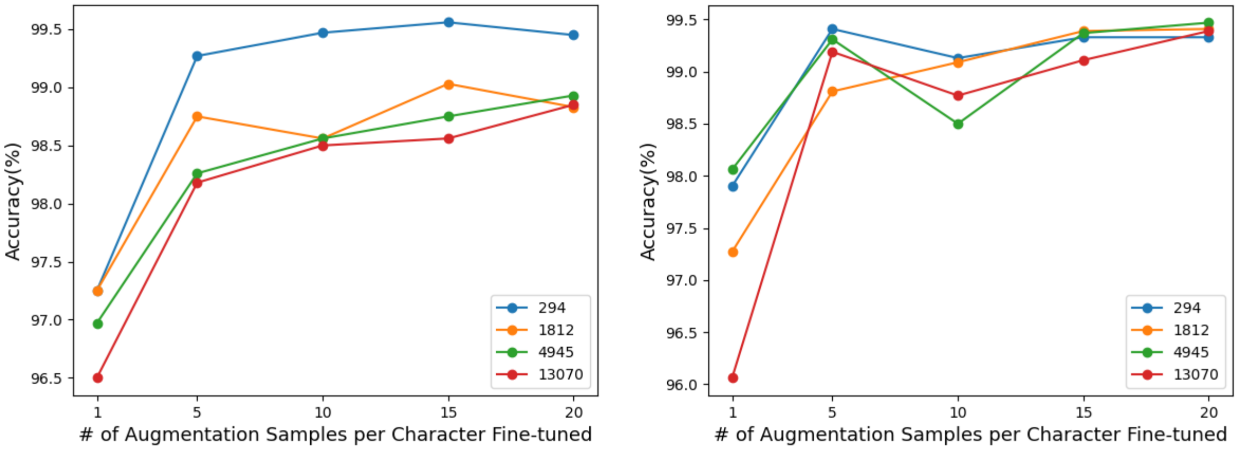
4.3.2. Transfer Learning Model

One of the main advantages of transfer learning is that a pre-trained model can quickly adapt to new data through fine-tuning training using small amounts of real data. To evaluate the performance of transfer learning, we fine-tuned various classification models with different amounts of real data by providing 10 and 20 augmentation samples per character. However, if only a small number of characters are used to fine-tune large-scale character classification, the model becomes biased towards a small number of characters and thus loses the recognition rate for other characters. To avoid deviations, data balancing is applied by recruiting the synthetic data generated in Section 3.1 for the remaining characters and then performing training. Table 8 presents the results of fine-tuning using three different amounts of real character data (100, 200, and 294 characters) for five training scenarios. We supplied varying numbers of augmented samples per character to perform transfer learning (namely 1, 5, 10, 15, and 20), and subsequently, assessed the recognition performance. After the data were generated and balanced, the problems of insufficient data and learning deviation were solved simultaneously. Table 8 illustrates that the data augmentation improved the recognition outcomes, with boldface indicating the best performance. As may be observed from Table 8, our transfer model was able to improve the recognition accuracy of characters on printed ID cards with fine-tuning. Figure 11 shows that the character recognition performance was dramatically improved when the models were fine-tuned using more real data. Figure 12 shows that the character recognition performance was improved when the models were fine-tuned using more augmentation data per character.



(**a**) (**b**)

Figure 11. Character recognition performance was improved dramatically when the models were fine-tuned using more numbers of real character data (100, 200, and 294) for two training scenarios, namely, (**a**) 5 augmentation samples per character, and (**b**) 20 augmentation samples per character.



(**a**) (**b**)

Figure 12. Character recognition performance was improved when the models were fine-tuned using more numbers of augmentation data (5, 10, 15, and 20) for two training scenarios, namely, (**a**) 200 real characters, and (**b**) 294 real characters.

Table 8. Testing results on transfer model fine-tuned using three different amounts of real character data (100, 200, and 294 characters) for five training scenarios, namely, 1, 5, 10, 15, and 20 augmentation samples per character.

# of Augmentation Samples per Character	# of Output Classes	# of Characters Fine-tuned in Transfer Learning			
		0	100	200	294
1	294	90.62%	96.91%	97.25%	97.90%
	1812	88.21%	97.31%	97.25%	97.27%
	4945	89.99%	95.79%	96.97%	98.06%
	13,070	83.72%	95.00%	96.50%	96.06%
5	294	90.62%	**99.25%**	99.27%	**99.41%**
	1812	88.21%	**98.89%**	98.75%	**99.81%**
	4945	89.99%	98.40%	98.26%	99.31%
	13,070	83.72%	97.61%	98.18%	99.19%
10	294	90.62%	99.13%	99.47%	99.13%
	1812	88.21%	98.81%	98.56%	99.09%
	4945	89.99%	98.00%	98.56%	98.50%
	13,070	83.72%	98.16%	98.50%	98.77%
15	294	90.62%	98.85%	**99.56%**	99.33%
	1812	88.21%	98.85%	**99.03%**	99.39%
	4945	89.99%	98.26%	98.75%	99.37%
	13,070	83.72%	**98.34%**	98.56%	99.11%
20	294	90.62%	98.75%	99.45%	99.33%
	1812	88.21%	98.58%	98.83%	99.41%
	4945	89.99%	**98.87%**	**98.93%**	**99.47%**
	13,070	83.72%	97.34%	**98.85%**	**99.39%**

4.4. Error Analysis and Performance Enhancement

In Section 4.3, we demonstrate that the two proposed transfer learning methods were able to strengthen the model trained using synthesized data and real data. The experimental results show that the recognition rate was improved, and it reached more than 95% after adding a few real samples for transfer learning. Although our model achieved good recognition results under large-scale classifications, some issues remain nonetheless. Figure 13 shows some samples with incorrect recognition in the large-scale classification of 13,070 characters. There are some Chinese characters that seem to be similar however, in reality, they have completely different meanings. On ID cards with a cluttered background, the text is more likely to be disturbed by background noise. In addition, an ID card may be stained or damaged due to wear, which may result in misclassifications by the CNN model. Figure 13a displays two examples of misclassification, as mentioned above. In addition, if the text segmentation method is not sufficiently robust, it may easily cause the cropped text image to contain more than one character, or to retain only parts of the character. Figure 13b presents two examples of misclassification due to inappropriate cropping.

To solve the recognition problem caused by character segmentation shifting, we utilized the projection method. First, the grayscale character image was converted to a binary image with a specific threshold to separate the character from the background. Then, binary pixels were projected in the horizontal and vertical directions separately to locate the area of the character in the image. However, the ID card images may include a noisy background, and the projection may be affected by noise disturbances. Therefore, we performed morphology processing using a 1 × 7 vertical bar-type erosion operator on the horizontal projection image and disconnected the small adjacent blocks. Next, we search to check if there was a continuous black area from the boundary toward the center, as indicated by the red lines in Figure 14. Meanwhile, the distance between the two red lines was required to be greater than 60% of the original width or height to enclose a complete character. As shown in Figure 14, a better segmentation result was obtained. Next, the OCR system was re-evaluated using the rectified dataset.

Table 9 presents the new evaluation results of the proposed models with the 13,070 character classification task. It may be observed that the recognition accuracy was significantly improved after projection rectification. The model performance was further improved by incorporating the rectified real dataset into the learning process for both the mixed data model and the transfer learning model.

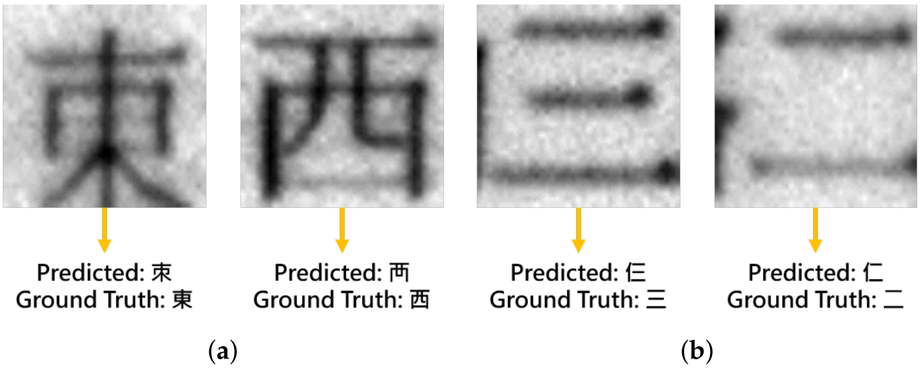


Figure 13. Some error misclassification samples: (**a**) Contaminated and similar error (Ground Truth: "East" and Ground Truth: "West") and (**b**) cropping error (Ground Truth: "Three" and Ground Truth: "Two").

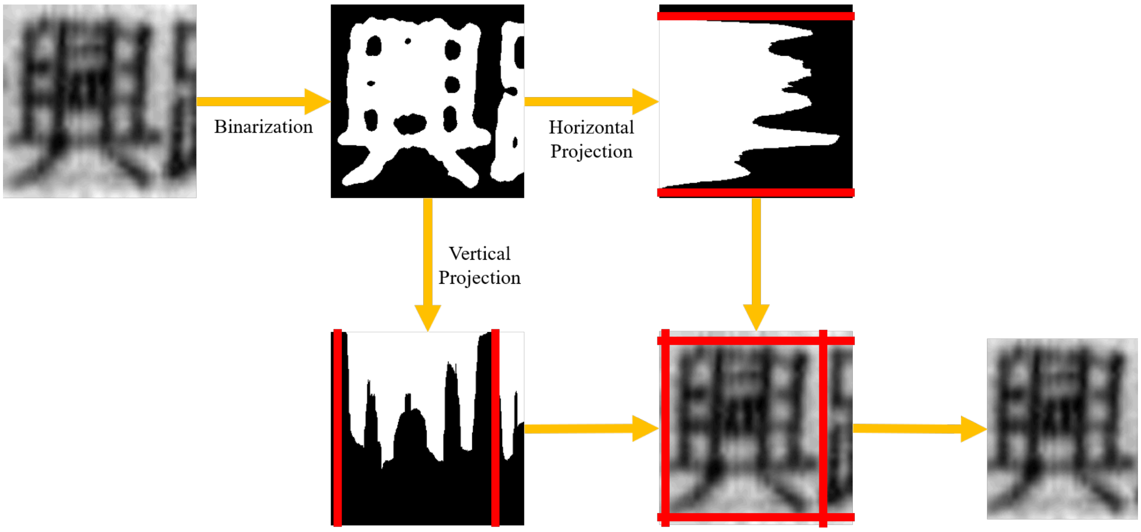


Figure 14. Procedure of projection method.

Table 9. New evaluation results of the proposed models on the 13,070 characters classification task using the projection rectification testing set.

Model	Training Samples	Original	Projection
Font model	600	83.72%	**85.70%**
Mixed model	600	91.95%	**96.54%**
Transfer model	400	95.01%	**97.53%**

4.5. Performance Comparison

Using our real ID card character testing dataset, we compared our proposed approach to EasyOCR [19] and PadddleOCR [18], and evaluated the recognition performance. EasyOCR and PaddleOCR exhibited recognition accuracies of 55.22% and 0.85%, respectively. Our system had a 99.39% success rate. Despite the apparent disparity, the input image formats were different. EasyOCR and PaddleOCR takes an image of a line of text as input, whereas our approach takes an image of a single character as input to perform recognition on each character sequentially. A single-character picture collected from real IDs forms the basis of our testing dataset. The evaluation and performance comparison may be skewed, as few alternative methods for recognizing printed Chinese characters on ID cards have been reported in the literature. To the best of our knowledge, Wang et al.'s most recent work achieved an average character recognition accuracy of 98.57%. Our proposed technique outperformed their approach (99.39% vs. 98.57%). The testing dataset, however, differed notably. Hence, making a comparison is difficult owing to the lack of a consistent baseline to evaluate the benefits of new technologies.

5. Conclusions

The remarkable and rapid development of deep learning technology has achieved great successes in computer vision and has also played a pivotal role in related fields. In this study, we developed an automatic OCR system for identifying up to 13,070 large-scale printed Chinese characters by using deep learning neural networks and fine-tuning techniques. The proposed framework comprises four components: Training dataset synthesis, background simulation, image preprocessing and data augmentation, model training, and transfer learning. Specifically, the training data synthesis procedure was composed of character font generation and a background simulation process. Three background models were proposed to simulate the background noise patterns on ID cards. Subsequently, the character font text is pasted on various backgrounds to generate character sample images. The preprocessing and data augmentation module first performs the min-max normalization operation to consistently rescale the brightness of the character images. Then, rotation and zooming data augmentation were applied to the synthesized training dataset to expand the diversity. A massive dataset of more than 19.6 million images was created to accommodate the variations of input images and strengthen the learning capacity of the CNN model. The proposed data generation method was validated experimentally to simulate the text data on ID cards and use a consistent normalization process to improve the brightness and contrast of the original image, making the model more adaptable to different backgrounds on ID card characters. In the deep learning network design, we modified GoogLeNet by replacing the FC layer with a GAP layer to avoid overfitting caused by a large amount of training data. The number of model parameters was consequently reduced. Finally, we employed the transfer learning technique to further refine the CNN model using a very small number of real data samples. The two transfer learning models we proposed can improve the learning of the original model within an acceptable range. Through the usage of real data, the proposed approach can be adapted to the characters on ID cards. Furthermore, the input character images were further rectified by applying the projection method. After the implementation of the data balance, transfer learning can be performed from each category on average, instead of only targeting a few characters, which considerably reduces the instability caused by the use of transfer learning under a large-scale classification. Overall, the overall recognition performance improved significantly. The experimental results demonstrate that the proposed framework is effective, and the accuracy of the large-scale 13,070 character recognition system was as high as 99.39% when evaluated on our real ID card dataset.

Although our model exhibited good results on the identification of characters on ID cards, certain characters were nonetheless recognized incorrectly; in particular, there are quite a few Chinese characters that seem to be similar, but in fact they are not the same, and their meanings are quite different. In addition, there is still a certain degree of difference between the characters on the ID cards and the simulated data generated by us. In future, we hope to collect more samples of Chinese characters from ID cards for fine-tuning training. We are currently deploying the proposed OCR framework to a real-time identification device and mobile phone, aiming to reduce labor demand in the service industry. Accurate automated identification can also reduce the possibility of human error. In addition, we intend to compare the accuracy of current state-of-the-art methods with our proposed method under various environments and operations; we also aim to improve the objective validation of our method and its OCR performance. Transformer-based approaches such as SpectralFormer [41], DETR [42], ViT [43], and ViT variants [44] have been found to be effective on image classification tasks. Hence, we intend to examine the feasibility of implementing a transformer-based solution for ID card printed character recognition in future work. Adversarial instances have also been used to test deep neural networks. We plan to combine the two types of adversarial methods presented by Kwon and Baek [45], and develop adversarial examples of ID card characters that are difficult for humans to recognize so as to test the robustness and recognition performance of the OCR system created. Lastly, the exploration of additional language systems would also be beneficial, as would designing the system to recognize all types of ID card text.

Author Contributions: Conceptualization, Y.-Q.L., H.-S.C. and D.-T.L.; Data curation, Y.-Q.L. and H.-S.C.; Formal analysis, Y.-Q.L. and D.-T.L.; Funding acquisition, D.-T.L. Investigation, Y.-Q.L. and D.-T.L.; Methodology, Y.-Q.L., H.-S.C. and D.-T.L.; Project administration, Y.-Q.L.; Resources, Y.-Q.L. and H.-S.C.; Software, H.-S.C.; Validation, Y.-Q.L. and H.-S.C.; Writing—original draft, H.-S.C.; Writing—review and editing, D.-T.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by the Ministry of Science and Technology, Taiwan, grant MOST 110-2622-E-305-001 and by the Orbit Technology Incorporation.

Data Availability Statement: The pre-trained model, dataset synthesizer source code, and datasets presented in this paper are available at http://imslab.csie.ntpu.edu.tw/index.php/dataset upon request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Aprillian, H.D.D.; Purnomo, H.D.; Purwanto, H. Utilization of Optical Character Recognition Technology in Reading Identity Cards. *Int. J. Inf. Technol. Bus.* **2019**, *2*, 38–46.
2. Purba, A.M.; Harjoko, A.; Wibowo, M.E. Text Detection In Indonesian Identity Card Based On Maximally Stable Extremal Regions. *Indones. J. Comput. Cybern. Syst.* **2019**, *13*, 177–188. [CrossRef]
3. Satyawan, W.; Pratama, M.O.; Jannati, R.; Muhammad, G.; Fajar, B.; Hamzah, H.; Fikri, R.; Kristian, K. Citizen Id Card Detection using Image Processing and Optical Character Recognition. In *Journal of Physics: Conference Series*; IOP Publishing: Bristol, UK, 2019; Volume 1235, p. 012049.
4. Tavakolian, N.; Nazemi, A.; Fitzpatrick, D. Real-time information retrieval from Identity cards. *arXiv* **2020**, arXiv:2003.12103.
5. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1–9. [CrossRef]
6. Xu, Y.; Shan, S.; Qiu, Z.; Jia, Z.; Shen, Z.; Wang, Y.; Shi, M.; Eric, I.; Chang, C. End-to-end subtitle detection and recognition for videos in East Asian languages via CNN ensemble. *Signal Process. Image Commun.* **2018**, *60*, 131–143. [CrossRef]
7. Zhong, Z.; Jin, L.; Xie, Z. High performance offline handwritten Chinese character recognition using GoogLeNet and directional feature maps. In Proceedings of the 2015 13th International Conference on Document Analysis and Recognition (ICDAR), Tunis, Tunisia, 23–26 August 2015; pp. 846–850. [CrossRef]
8. Yin, F.; Wang, Q.F.; Zhang, X.Y.; Liu, C.L. ICDAR 2013 Chinese Handwriting Recognition Competition. In Proceedings of the 2013 12th International Conference on Document Analysis and Recognition, Washington, DC, USA, 25–28 August 2013; pp. 1464–1470. [CrossRef]
9. Lin, M.; Chen, Q.; Yan, S. Network in network. *arXiv* **2013**, arXiv:1312.4400.
10. Li, Z.; Teng, N.; Jin, M.; Lu, H. Building efficient CNN architecture for offline handwritten Chinese character recognition. *Int. J. Doc. Anal. Recognit.* **2018**, *21*, 233–240. [CrossRef]
11. Xiao, X.; Jin, L.; Yang, Y.; Yang, W.; Sun, J.; Chang, T. Building fast and compact convolutional neural networks for offline handwritten Chinese character recognition. *Pattern Recognit.* **2017**, *72*, 72–81. [CrossRef]
12. Melnyk, P.; You, Z.; Li, K. A high-performance CNN method for offline handwritten Chinese character recognition and visualization. *Soft Comput.* **2019**, *24*, 7977–7987. [CrossRef]
13. Su, Y.S.; Chou, C.H.; Chu, Y.L.; Yang, Z.Y. A Finger-Worn Device for Exploring Chinese Printed Text With Using CNN Algorithm on a Micro IoT Processor. *IEEE Access* **2019**, *7*, 116529–116541. [CrossRef]
14. Yao, L.; Mao, C.; Luo, Y. Graph convolutional networks for text classification. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 7370–7377.

15. Liu, B.; Xu, X.; Zhang, Y. Offline Handwritten Chinese Text Recognition with Convolutional Neural Networks. *arXiv* **2020**, arXiv:2006.15619.
16. Wang, J.; Wu, R.; Zhang, S. Robust Recognition of Chinese Text from Cellphone-acquired Low-quality Identity Card Images Using Convolutional Recurrent Neural Network. *Sens. Mater.* **2021**, *33*, 1187–1198. [CrossRef]
17. Du, Y.; Li, C.; Guo, R.; Yin, X.; Liu, W.; Zhou, J.; Bai, Y.; Yu, Z.; Yang, Y.; Dang, Q.; et al. PP-OCR: A practical ultra lightweight OCR system. *arXiv* **2020**, arXiv:2009.09941.
18. PaddleOCR. Available online: https://github.com/PaddlePaddle/PaddleOCR (accessed on 26 May 2021).
19. EasyOCR. Available online: https://github.com/JaidedAI/EasyOCR (accessed on 11 September 2021).
20. Shi, B.; Bai, X.; Yao, C. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 2298–2304. [CrossRef]
21. Graves, A.; Fernández, S.; Gomez, F.; Schmidhuber, J. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In Proceedings of the 23rd International Conference on Machine Learning, Pittsburgh, PA, USA, 25–29 June 2006; pp. 369–376.
22. Chen, P.C. Traditional Chinese Handwriting Dataset. 2020. Available online: https://github.com/AI-FREE-Team/Traditional-Chinese-Handwriting-Dataset (accessed on 19 December 2021).
23. Xu, Y.; Yin, F.; Wang, D.H.; Zhang, X.Y.; Zhang, Z.; Liu, C.L. CASIA-AHCDB: A Large-Scale Chinese Ancient Handwritten Characters Database. In Proceedings of the 2019 International Conference on Document Analysis and Recognition (ICDAR), Sydney, Australia, 20–25 September 2019; pp. 793–798. [CrossRef]
24. Zhong, Z.; Jin, L.; Feng, Z. Multi-font printed Chinese character recognition using multi-pooling convolutional neural network. In Proceedings of the 2015 13th International Conference on Document Analysis and Recognition (ICDAR), Tunis, Tunisia, 23–26 August 2015; pp. 96–100. [CrossRef]
25. Qiu, S. Global weighted average pooling bridges pixel-level localization and image-level classification. *arXiv* **2018**, arXiv:1809.08264.
26. Zhang, Q.; Lee, K.C.; Bao, H.; You, Y.; Li, W.; Guo, D. Large Scale Classification in Deep Neural Network with Label Mapping. In Proceedings of the 2018 IEEE International Conference on Data Mining Workshops (ICDMW), Singapore, 17–20 November 2018; pp. 1134–1143. [CrossRef]
27. Zhang, J.; Zhu, Y.; Du, J.; Dai, L. Radical Analysis Network for Zero-Shot Learning in Printed Chinese Character Recognition. In Proceedings of the 2018 IEEE International Conference on Multimedia and Expo (ICME), San Diego, CA, USA23–27 July 2018; pp. 1–6. [CrossRef]
28. Zhang, J.; Du, J.; Dai, L. Radical analysis network for learning hierarchies of Chinese characters. *Pattern Recognit.* **2020**, *103*, 107305. [CrossRef]
29. Wu, C.; Wang, Z.R.; Du, J.; Zhang, J.; Wang, J. Joint Spatial and Radical Analysis Network For Distorted Chinese Character Recognition. In Proceedings of the 2019 International Conference on Document Analysis and Recognition Workshops (ICDARW), Sydney, Australia, 22–25 September 2019; Volume 5, pp. 122–127. [CrossRef]
30. Hong, D.; Gao, L.; Yokoya, N.; Yao, J.; Chanussot, J.; Du, Q.; Zhang, B. More diverse means better: Multimodal deep learning meets remote-sensing imagery classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 4340–4354. [CrossRef]
31. Qiao, S.; Liu, C.; Shen, W.; Yuille, A. Few-Shot Image Recognition by Predicting Parameters from Activations. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7229–7238. [CrossRef]
32. Ao, X.; Zhang, X.Y.; Yang, H.M.; Yin, F.; Liu, C.L. Cross-Modal Prototype Learning for Zero-Shot Handwriting Recognition. In Proceedings of the 2019 International Conference on Document Analysis and Recognition (ICDAR), Sydney, Australia, 20–25 September 2019; pp. 589–594. [CrossRef]
33. Tang, Y.; Peng, L.; Xu, Q.; Wang, Y.; Furuhata, A. CNN Based Transfer Learning for Historical Chinese Character Recognition. In Proceedings of the 2016 12th IAPR Workshop on Document Analysis Systems (DAS), Santorini, Greece, 11–14 April 2016; pp. 25–29. [CrossRef]
34. Tang, Y.; Wu, B.; Peng, L.; Liu, C. Semi-Supervised Transfer Learning for Convolutional Neural Network Based Chinese Character Recognition. In Proceedings of the 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 9–15 November 2017; Volume 1, pp. 441–447. [CrossRef]
35. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]
36. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
37. Clark, A. Pillow (PIL Fork) Documentation. 2015. Available online: https://pillow.readthedocs.io/en/stable/ (accessed on 19 December 2021).
38. Redmon, J.; Farhadi, A. YOLOv3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
39. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.A. Inception-v4, inception-resnet and the impact of residual connections on learning. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
40. Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision (ECCV)*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 818–833.

41. Hong, D.; Han, Z.; Yao, J.; Gao, L.; Zhang, B.; Plaza, A.; Chanussot, J. Spectralformer: Rethinking hyperspectral image classification with transformers. *IEEE Trans. Geosci. Remote Sens.* **2021**. [CrossRef]
42. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 213–229.
43. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
44. Han, K.; Wang, Y.; Chen, H.; Chen, X.; Guo, J.; Liu, Z.; Tang, Y.; Xiao, A.; Xu, C.; Xu, Y.; et al. A survey on visual transformer. *arXiv* **2020**, arXiv:2012.12556.
45. Kwon, H.; Baek, J.W. Adv-Plate Attack: Adversarially Perturbed Plate for License Plate Recognition System. *J. Sens.* **2021**, *2021*, 6473833. [CrossRef]