Deep Neural Networks for Handwritten Chinese Character Recognition

Renan G. Maidana*, Juarez Monteiro*, Roger Granada*, Alexandre M. Amory† and Rodrigo C. Barros†
Faculdade de Informática
Pontifícia Universidade Católica do Rio Grande do Sul
Av. Ipiranga, 6681, 90619-900, Porto Alegre, RS, Brazil
* Email: {renan.maidana, juarez.santos, roger.granada}@acad.pucrs.br
† Email: {alexandre.amory, rodrigo.barros}@pucrs.br

***Abstract*—Automatic handwriting recognition is an important task since it can be used to replace human beings in various activities such as identifying postal addresses on envelopes, information in bank checks, and several other tedious tasks that humans need to perform. Convolutional Neural Networks are a power machine learning method for computer vision tasks, having achieved state-of-the-art results in the recognition of handwritten Arabic digits and also in multiple distinct alphabets. In this work, we extensively explore the performance of those networks for handwritten Chinese characters recognition (HCCR). For such, we have trained several models based on popular convolutional neural networks architectures that are commonly used for large-scale image recognition, and we also employ several distinct architectural fusion methods, resulting in more than 18 classification approaches. We report the results of all 18 configurations in the well-known HWDB and ICDAR2013 datasets.**

I. Introduction

Automatic handwriting recognition (HWR) is the task of transforming a language represented in its spatial form of graphical marks into its symbolic representation [1]. This task covers many types of applications, including reliable person authentication [2], writer identification [3], and handwritten digit/character/word recognition [4], [5], [6]. HWR is the process whose objective is to interpret and identify any character in the handwritten form.

Techniques of HWR can be broadly categorized as either online (stroke trajectory-based) or offline (image-based), based on the availability of dynamic information. The former usually involves the automatic conversion of the text as it is written on a digitizer, using temporal spatial information generated from the movement of a stylus on an electromagnetic surface (*e.g.*, velocity, movement direction, number of strokes, *etc.*). The latter involves the automatic conversion of the text from static images captured by optical devices (such as a cameras or scanners) into letter codes by the computer. Due to the limited information in offline recognition systems, their accuracy tends to be lower when compared to online recognition systems [1].

Although handwritten character recognition has received intensive attention in the last decades, it still remains a challenging problem due to the the presence of cursive writing, touching strokes, and confusion in shapes [7]. Compared to the task of recognizing handwritten digits and letters in Latin alphabets, Handwritten Chinese Character Recognition (HCCR) is a more challenging task mainly due to two reasons. First, Chinese characters are ideographic in nature, with more than 50,000 characters of which 6,000 are commonly used and have a wide range of complexity [8], whereas English has only 26 characters. Second, most Chinese characters have much more complicated structures and consist of much more strokes compared to arabic digits or English characters. Moreover, some Chinese characters have similar characteristics and very different meanings, as illustrated in Figure 1.

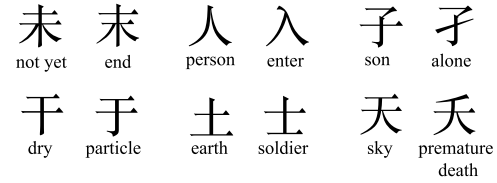

Fig. 1. Chinese characters with similar forms and different meanings.

Recent approaches try to solve the problems of HWR using deep neural networks such as Convolutional Neural Networks (CNNs) [3], [9], [7]. In this paper, we also propose the use of CNN for offline HCCR in images. Our study differs from previous work by being the first to exhaustively test several well-known CNN architectures and also strategies for fusing their performance. For such an analysis, we train and test the networks using the HWDB1.1 version of the Institute of Automation of Chinese Academy of Sciences (CASIA) dataset, and results show that CNNs surpass human accuracy in this task.

The rest of this paper is structured as follows. Section II reports work that also employ machine learning for identifying handwritten characters. In Section III, we describe the architectures we use to recognize handwritten Chinese characters. Section IV describes our experimental settings and the corresponding results. Finally, the paper ends with our conclusions and future work directions in Section VI.

II. Related Work

HCCR has been a topic of discussion in pattern recognition before convolutional networks became widely popular, although there were cases of CNN usage in HCCR dating

978-1-5386-2407-4/17 $31.00 © 2017 IEEE
DOI 10.1109/BRACIS.2017.24

back to 1993. In [6], the authors used a LeNet to classify a limited Chinese vocabulary, achieving test accuracy of 94.2%. Arguably, the limited size of the vocabulary was due to limited computational resources, as even today training and testing a convolutional network is not a trivial task.

Simultaneously with the rise of CNNs, traditional methods of HCCR were being used, which rely on the identification and extraction of specific features such as shape or stroke density. For example, Tang *et al.* [10] perform offline HCCR by combining several feature-based classifiers, which use peripheral shape, stroke density, and stroke direction information, achieving an accuracy of 90% on a large dataset for the time, with 540,100 samples.

The first use of CNN for HCCR in a large dataset was by Ciresan *et al.* [11], with a multi-column deep neural network to classify the characters from the CASIA dataset, among other tasks in image recognition (*e.g.* traffic signs recognition). On Offline HCCR, the authors obtained an error rate of 6.5%.

The NLPR also held several competitions for Chinese Handwriting Recognition, the most prominent being from ICDAR-2013. In offline recognition, the team from Fujitsu Center took the prize with a 4-CNN voting system. The CNN themselves were composed of ten convolutional layers and two fully connected layers, trained in the HWDB1.1 dataset [12]. The ICDAR 2013 dataset is used nowadays to train and evaluate HCCR solutions.

Cheng *et al.* [13] presented a deep triplet network (DTN) method, which learns a CNN model using both classification and similarity ranking signals as supervision. It maximizes the inter-class variations, minimizes the intra-class variations, and simultaneously minimizes the cross-entropy loss. The work by Zhong *et al.* [14] uses directional feature maps along with an ensemble of AlexNet and GoogLeNet [15] architectures. The paper reports classification with 96.74% accuracy in the ICDAR 2013 offline test set. We did not include them in our results since we are using a reduced version of ICDAR data due to limitations of computational power.

Finally, Zhang *et al.* [16] boost the performance of their CNN by applying shape normalization and direction decomposition on the images to extract feature maps (or "directMaps"), with a technique known as *Normalization-cooperated Gradient Feature Extraction* [17]. Apart from being domain-specific knowledge of HCCR, the directMaps eliminates the need of data augmentation and model ensemble. The authors also implemented an adaptive layer to the CNN for compensating handwriting variation. The layer reduces mismatch between training and test data from a writer, caused by different handwriting styles across the writer's images. The *directMap + CNN + Adaptive Layer* architecture achieves 97.37% accuracy on the full ICDAR 2013 test set, which is to the best of our knowledge the current state-of-the-art of HCCR.

III. Convolutional Neural Networks

Handwriting recognition has been the focus of much research in the last decades. In particular, the application of convolutional neural networks in handwriting recognition started in the early 90s [18], [6], with the pioneering work of Yann LeCun. Due to dramatic advances in hardware and greater availability of annotated data through crowdsourcing initiatives, CNNs have become very popular for classifying images, outperforming the previous approaches based on handcrafted features by large margins (see, e.g., the work of Krizhevsky et al. [19] for large-scale image classification).

The intuition behind CNNs is to perform automatic representation learning, transforming the raw data into a set of features (i.e., latent space) that is well-suited to discriminate the concepts needed for detection or classification. This latent space is represented as a nested hierarchy of concepts, with each concept defined in relation to simpler concepts, and more abstract representations computed in terms of less abstract ones [18].

In this paper we replicate several well-known award-winning CNN architectures, namely LeNet [18], AlexNet [19], ZFNet [20], VGG-5 [21], VGG-7 [21], VGG-9 [21], and VGG-16 [21]. Our hypothesis is that networks with different architectures may identify different features in images, and that by combining (or fusing) such networks we may be capable of improving results, which may be viewed as a particular kind of *ensemble effect*. Table I presents the architecture of each network that is employed in our experimental scenario. All architectures use batch normalization [22] after each convolutional layer, omitted in Table I for brevity. Explanation on each architecture is as follows.

- **LeNet** is an architecture proposed by LeCun *et al.* [18] and contains two (5×5) convolutional layers followed by (2×2) pooling layers. Two Fully-Connected (FC) layers connect all activations in the previous layer and a *softmax* function is applied in the last layer to generate class probabilities. This network was the first successful application of CNNs in real-world problems, and it was initially developed to identify digits and zip codes from images.
- **AlexNet** is an architecture proposed by Krizhevsky *et al.* [19] and consists of a (11×11) convolutional layer followed by other three (3×3) convolutional layers. The network contains two FC layers followed by the *softmax* function in the output layer. This architecture became well-known after wining the ImageNet Large Scale Visual Recognition Competition (ILSVRC) in 2012, dramatically reducing the top-5 test error rate when compared with previous results.
- **ZFNet** is an architecture proposed by Zeiler and Fergus [20] and contains five convolutional layers: two layers containing (7×7) and (5×5) filters respectively, and three layers containing (3×3) filters. Max pooling is performed between some of the convolutional layers and two FC layers followed by the *softmax* function complete the network. This architecture is an improvement over AlexNet that modifies a few of the network's hyperparameters, in particular by expanding the size of the middle convolutional layers and making the stride and filter size on the first layer smaller.

TABLE I
CNN ARCHITECTURES INVESTIGATED IN THIS WORK.

LeNet	ZFNet	VGG-5	VGG-7	VGG-9	VGG-16	AlexNet
input data (64×64 images)						
conv-64	conv-96	conv-64	conv-64	conv-64	conv-64 conv-64	conv-96
maxpooling						
conv-128	conv-96	conv-128	conv-128	conv-128	conv-128 conv-128	conv-128 conv-128
maxpooling						
FC-4096 FC-200	conv-256 conv-384 conv-384	conv-256	conv-256	conv-256 conv-256	conv-256 conv-256 conv-256	conv-384
	maxpooling					
	FC-4096 FC-4096 FC-200	FC-4096 FC-200	conv-512	conv-512 conv-512	conv-512 conv-512 conv-512	conv-192 conv-192
			maxpooling			
			FC-4096 FC-4096 FC-200	FC-4096 FC-4096 FC-200	conv-512 conv-512 conv-512	conv-128 conv-128
					maxpooling	
					FC-4096 FC-4096 FC-200	FC-4096 FC-4096 FC-200

- **VGG-5**, **VGG-7**, **VGG-9**, and **VGG-16** are modified versions of the networks developed by Simonyan and Zisserman [21] from Oxford's renowned Visual Geometry Group (VGG). The VGG architecture contains a stack of (3×3) convolutional layers and 2×2 pooling layers (not all the convolutional layers are followed by max-pooling). The stack is followed by FC layers and the *softmax* function in the output layer. The VGGs used here differ in the number of convolutional and FC layers, with the number indicating the number of total weight layers of the architecture.

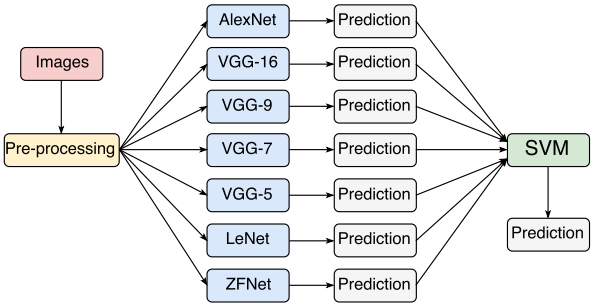

Fig. 2. Pipeline of the fusion architecture.

A. Pre-processing and Network Fusion

Figure 2 presents the fusion pipeline, in which we apply *pre-processing* before sending the images to the CNNs described in Table I. The pre-processing step consists of extracting and converting the data to an image format, since the original format is proprietary. Each image is resized to 56×56 pixels and a white border or 4 pixels is added to each margin in order to centralize the character and avoid strokes close to the border. Finally, the image is enhanced via contrast stretching [23], which attempts to improve the contrast in an image by "stretching" the range of intensity values it contains to span a desired range of values.

Each network $N_j \in \mathbf{N}$ (where $\mathbf{N}$ is the set of all CNNs in Table I) processes data and generates probability scores based on the *softmax* function, i.e., the probability $p(C_i|\mathbf{x}; N_j)$ that N_j has of classifying image $\mathbf{x}$ as belonging to class C_i. Using the *softmax* scores, we train a Support Vector Machine (SVM) [24] in order to perform a late fusion of the networks. The SVM fusion method was chosen because it is commonly used and consistently presents good results in the literature. Similarly to the *softmax* function, the late fusion generates a vector containing the probabilities $p(C_i|\mathbf{x}; \mathcal{F}(\mathbf{N}))$ of classifying image $\mathbf{x}$ as belonging to each class based on the fusion of networks $\mathcal{F}(\mathbf{N})$.

IV. EXPERIMENTAL ANALYSIS

In this section, we describe the dataset used in our experiments, the implementation details regarding the convolutional neural networks, and the results achieved by our approach in comparison with the current state-of-the-art.

A. Dataset

The CASIA[1] Online and Offline Chinese Handwriting Databases [25] were developed by the National Laboratory of Pattern Recognition (NLPR), Institute of Automation of Chinese Academy of Sciences (CASIA), and contains versions for online (stroke trajectories) and offline (isolated images) handwritten characters, named OLHWDB and HWDB respectively. The isolated character samples are divided into three datasets, and involve 7,356 character classes, including 7,185 Chinese characters and 171 alphanumeric and symbols.

[1] http://www.nlpr.ia.ac.cn/databases/handwriting/Home.html

The offline HWDB1.1 version was produced by 300 writers and contains 3,755 classes, divided into 897,758 samples for training (240 writers) and 223,991 samples for testing (60 writers). In our experiments, we split the original training dataset into training and validation sets. The training set contains 200 writers and the validation set the remaining 40 writers.

Since HWDB1.1 is a rather large dataset, we perform experiments using a subset containing 200 classes (different Chinese characters) divided into 33,258 samples for training, 9,593 for validation, and 4,790 for test. This new subset enables a faster training of the CNNs, allowing us to properly compare them and find which architecture is the most adequate to this task. The rationale is that if the network can achieve a good performance using a random subset of the data, it should also achieve good results using the entire dataset. This subset containing 200 classes is referred hereafter as the *200-Class* set.

Besides HWDB1.1, we perform experiments using the *ICDAR2013* [12] dataset, though only as a test set. *ICDAR2013* is a dataset available for the Chinese Handwriting Recognition Competition, for online and offline character recognition, held with the 2013 edition of the International Conference on Document Analysis and Recognition (ICDAR). Similarly to HWDB1.1, *ICDAR2013* contains the same 3,755 classes distributed in 60 writers with 225,300 images. Similarly to what was done to HWDB1.1, we also randomly defined a subset with the same 200-classes from *ICDAR2013*. This subset, hereafter simply called *ICDAR2013*, is the one used in the experimental analysis.

B. Settings and Hyper-Parameter Definition

For model fitting, all architectures (*LeNet*, *AlexNet*, *ZFNet*, *VGG-5*, *VGG-7*, *VGG-9* and *VGG-16*) are trained from scratch using the 200-Class (HWDB1.1) training set. Each network is trained for 20 epochs with batches containing 100 frames. We use the validation set to verify the model that performs best, *i.e.*, the model with the highest accuracy for each network during training.

Weights are optimized using *AdaDelta* [26] with learning rate $lr = 1$, $\rho = 0.95$, $\epsilon = 1 \times 10^{-8}$ and learning rate decaying 50% every 3 epochs. Fully-connected layers in all architectures are L2-norm regularized, with $\lambda = 0.0005$. We additionally regularize the networks by applying dropout on the fully-connected layers with a probability of 50%. All networks use rectified linear activation units (ReLU) (*relu*$(x) = max(x, 0)$).

Using the probabilities extracted from the CNNs from images of the validation data, we train a Support Vector Machine in order to perform the late fusion of the networks. The late fusion is trained with the probabilities of the best model selected via validation set. Thus, when performing the late fusion of *VGG-9* and *ZFNet* (*VGG9+ZFNet+SVM*), we concatenate the validation probabilities from both CNNs to train the SVM. In our experiments, we use the off-the-shelf implementation of a multi-class Support Vector Machine (*SVM*) by Crammer and Singer [24] from *scikit-learn*[2] toolbox. No kernel is used (ie.e, linear kernel) and default *scikit-learn* regularization parameter $C = 1$ is used, with the square of the *hinge loss* as loss function. No attempts were made to tune those hyper-parameters.

During the test phase, we forward each test instance throughout each network (instances from both test subsets *200-Class* and *ICDAR2013*). With the probabilities extracted from each network, we pass them to the SVM in order to generate the final classification. For instance, a forward pass on the *VGG-9* network followed by the SVM classification is named *VGG9+SVM*. For networks that undergo late fusion, a forward pass of each image is performed in each network, and the probabilities are then concatenated and passed to the SVM for the final classification. Hence, when classifying an image using the fusion of *VGG-9* and *ZFNet* (*VGG-9+ZFNet+SVM*), a forward pass of the image is performed in both *VGG-9* and *ZFNet*, and both vectors containing the probabilities are concatenated and passed to the SVM so it can perform the final classification.

V. Results

To evaluate the performance of the networks (individually and fused), we compare the accuracy of each CNN over the 200-Class test set, and also over the ICDAR2013 test set (reduced to 200-class). To analyze the performance of the fused networks, we use the classification of each individual CNN as a baseline, and thus we can see whether the SVM improves the predictive performance. Since we are evaluating seven baseline models, there are 13,699 possible CNN fusions.

Given this very large number, the fused CNN presented here were arbitrarily selected, reflecting what the authors believe to be the most promising among the possible candidates. The fused architectures are as follows:

- *VGG7+SVM*
- *VGG9+SVM*
- *VGG16+SVM*
- *ZFNet+SVM*
- *VGG7+VGG16+SVM*
- *VGG7+ZFNet+SVM*
- *VGG9+VGG16+SVM*
- *VGG9+ZFNet+SVM*
- *VGG16+ZFNet+SVM*
- *VGG9+VGG16+ZFNet+SVM*
- *VGG7+VGG9+VGG16+ZFNet+SVM*

Table II reports the results for the 7 individual networks as well as for the 10 fused architectures.

By training and testing our CNN implementations in the 200-class subset, we were able to identify which architecture was the most suited for the task of handwritten Chinese character recognition. Among the architectures, ZFNet achieves the highest accuracy in both 200-Class and ICDAR2013 datasets. The ZFNet architecture achieves the best global accuracy of 97.8% for *200-Class* and 98.2% for ICDAR2013. Recall that

[2]http://scikit-learn.org

TABLE II
ACCURACY FOR ALL BASELINES AND FUSION METHODS, TRAINED WITH THE SUBSET, FOR THE USED TEST SETS.

Approach	200-Class	ICDAR2013
LeNet	0.924	0.935
VGG5	0.943	0.956
VGG7	0.961	0.965
VGG9	0.967	0.972
VGG16	0.964	0.972
AlexNet	0.959	0.968
ZFNet	**0.978**	**0.982**
VGG7+SVM	0.961	0.966
VGG9+SVM	0.967	0.972
VGG16+SVM	0.964	0.972
ZFNet+SVM	**0.978**	**0.982**
VGG7+VGG16+SVM	0.969	0.974
VGG7+ZFNet+SVM	0.974	0.979
VGG9+VGG16+SVM	0.970	0.974
VGG9+ZFNet+SVM	0.970	0.979
VGG16+ZFNet+SVM	0.973	0.980
VGG9+VGG16+ZFNet+SVM	0.974	0.978
VGG7+VGG9+VGG16+ZFNet+SVM	0.975	0.978

ZFNet is an adaptation of AlexNet, increasing the size of the middle convolutional layers while tweaking a few of other hyperparameters. When comparing results from ZFNet and AlexNet, we can see that modifications in the architecture resulted in a great improvement for the task at hand. This result surpasses the 96.13% of human accuracy achieved in ICDAR2013 [16], meaning that the machine can identify Chinese characters better than humans do.

The worst results are achieved by LeNet network, which is expected since it is the oldest CNN architecture, with design choices that were not fully understood by the time of its publication. The depth of the LeNet's network is probably not sufficient to extract features that properly differentiate among Chinese characters. This characteristic is also observed in the VGG networks, since by increasing the number of layers the accuracy of the system also increases, except from VGG9 to VGG16, which may indicate a small level of overfitting of the larger architecture. Since VGG-9 and VGG-16 achieve almost the same values of accuracy, it probably indicates that keeping increasing the number of layers may not necessarily increase the number of important features to classify a character, though we did not evaluate the possibility of using residual connections [27] for allowing more depthness.

When comparing AlexNet to VGG networks, we see that it achieves an accuracy between the level of VGG-5 and VGG-9. Those results tend to confirm our assumption that shallow networks are not capable of properly solving the problem, while the larger networks may suffer from overfitting and should be more regularized. Possibly, some gradients on the deeper CNNs may still vanish, which have a negative effect on accuracy. Once again residual connections could be a possible solution for larger networks.

Adding SVMs as a post-processing classification step of the images does not improve the performance of most networks. However, SVM seems to be very important to use as fusing method that aggregates the performance two or more networks.

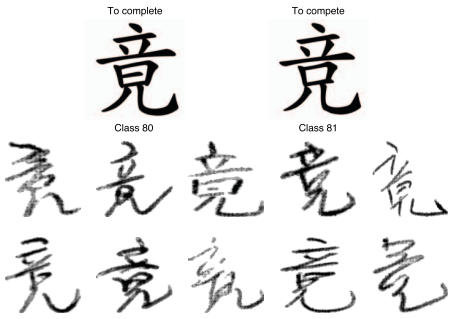

Fig. 3. *(Top)* Class 80 ("to complete") and class 81 ("to compete") represented by their respectively unicode character. *(Bottom)* 10 test images of class 80, which were classified as class 81.

For the fused networks, the best accuracy is achieved when merging ZFNet with other networks. On the other hand, there is no significant improvement relative to the stand-alone model, indicating that it is more interesting to keep a short architecture instead of mixing two or more different architectures. The non improvement in CNN fusions may be also an indicative that the CNNs are already at their maximum learning capability.

By analyzing the predictions per class for our best model, we were able to identify which class was most incorrectly predicted. Our CNNs tends to classify characters from class 80 as being from class 81. When checking the characters, we can understand that this misclassification may be due to the similarity between the two classes (the only difference is a horizontal stroke) and the wildly different handwriting styles and variations. To illustrate how similar classes 80 and 81 are, and the effect of different handwriting styles, Figure 3 depicts the two characters, representing classes 80 and 81, along with 10 samples of class 80 from the ICDAR2013 test set.

With regards to the results that are found in the literature, Handwritten Chinese character recognition was originally addressed by analyzing features specific to the handwriting of Chinese characters, commonly named today as *handcrafted features*. These methods achieve satisfactory results (*e.g.*, Tang *et al.* [10], with 90% accuracy), however they are not capable of reaching the level of deep learning approaches such as CNNs, since they tend to suffer more with data variation than learning representation approaches. Another problem is that they require specific knowledge on the problem to identify which features are significant for classification, and extensive experimentation to find out which of these features are the best for each instance. CNNs, on the other hand, are end-to-end approaches that require no human-intervention in the process of feature definition. They automatically learn to identify the most significant features for a given problem, and thus do not need any prior knowledge. In comparison, our best architecture surpasses the handcrafted state-of-the-art using a self-contained model, the ZFNet architecture.

VI. Conclusions and Future Work

In this work, we experimentally evaluated several deep learning architectures for handwritten recognition as well as a late fusion method. The pipeline of the architecture includes training CNNs in parallel to extract features (probabilities) from images and classify unseen images by a late fusion process. Using a subset of the Chinese characters from HWDB1.1 dataset, we perform experiments showing that the convolutional networks can indeed learn high-level relevant features for handwritten recognition.

We verified that a single convolutional neural network performs as good as multiple ones in the problem of classifying handwritten Chinese characters. We assume that a single network has sufficient power to extract all relevant features in order to classify them. When we experimented on multiple networks via late fusion, we observed that results did not improve as expected. A single ZFNet (97.8% accuracy) has shown better performance than 11 fusion combinations.

As future work, first we intend to train our CNNs using the full version of HWDB1.1 (3755 classes), assuming we will have enough computational power to perform those experiments. Second, we intend to augment our dataset using handcrafted features such as histogram of optical flow (HOF), histogram of gradients (HOG), motion boundary history (MBH), and dense trajectories to verify whether those features improve the classification of CNNs. Finally, as an application, we intend to build a translator app for smart-phones, in which: The image of a Chinese character is obtained through the phone's camera; The image's class is identified with our CNNs; The character is translated to English via a translation API (e.g. Google Cloud API); The identified Chinese character and its translation are shown to the user.

Acknowledgement

This paper was achieved in cooperation with HP Brasil Indústria e Comércio de Equipamentos Eletrônicos LTDA, using incentives of Brazilian Informatics Law (Law nº 8.2.48 of 1991).

References

[1] R. Plamondon and S. N. Srihari, "Online and off-line handwriting recognition: a comprehensive survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 63–84, 2000.

[2] M. Bashir and J. Kempf, "Person authentication with rdtw based on handwritten pin and signature with a novel biometric smart pen device," in *2009 IEEE Workshop on Computational Intelligence in Biometrics: Theory, Algorithms, and Applications*, 2009, pp. 63–68.

[3] M. I. Malik, M. Liwicki, L. Alewijnse, W. Ohyama, M. Blumenstein, and B. Found, "Icdar 2013 competitions on signature verification and writer identification for on- and offline skilled forgeries (sigwicomp 2013)," in *2013 12th International Conference on Document Analysis and Recognition*, 2013, pp. 1477–1483.

[4] A. Graves and J. Schmidhuber, "Offline handwriting recognition with multidimensional recurrent neural networks," in *Proceedings of the 21st International Conference on Neural Information Processing Systems*, ser. NIPS'08. USA: Curran Associates Inc., 2008, pp. 545–552.

[5] C.-L. Liu, F. Yin, D.-H. Wang, and Q.-F. Wang, "Online and offline handwritten chinese character recognition: Benchmarking on new databases," *Pattern Recognition*, vol. 46, no. 1, pp. 155–162, 2013.

[6] Q.-Z. Wu, Y. LeCun, L. D. Jackel, and B.-S. Jeng, "On-line recognition of limited-vocabulary chinese character using multiple convolutional neural networks," in *1993 IEEE International Symposium on Circuits and Systems*, 1993, pp. 2435–2438.

[7] C. Wu, W. Fan, Y. He, J. Sun, and S. Naoi, "Handwritten character recognition by alternately trained relaxation convolutional neural network," in *2014 14th International Conference on Frontiers in Handwriting Recognition*, 2014, pp. 291–296.

[8] W. Y. Leng and S. M. Shamsuddin, "Writer identification for chinese handwriting," *International Journal of Advances in Soft Computing and its Applications*, vol. 2, no. 2, pp. 142–173, 2010.

[9] D. Suryani, P. Doetsch, and H. Ney, "On the benefits of convolutional neural network combinations in offline handwriting recognition," in *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, 2016, pp. 193–198.

[10] Y. Y. Tang, L.-T. Tu, J. Liu, S.-W. Lee, and W.-W. Lin, "Off-line recognition of chinese handwriting by multifeature and multilevel classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 5, pp. 556–561, May 1998.

[11] D. C. Ciresan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," *CoRR*, vol. abs/1202.2745, 2012.

[12] F. Yin, Q.-F. Wang, X.-Y. Zhang, and C.-L. Liu, "Icdar 2013 chinese handwriting recognition competition," in *2013 12th International Conference on Document Analysis and Recognition*. IEEE, 2013, pp. 1464–1470.

[13] C. Cheng, X.-Y. Zhang, X.-H. Shao, and X.-D. Zhou, "Handwritten chinese character recognition by joint classification and similarity ranking," in *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, 2016, pp. 507–511.

[14] Z. Zhong, L. Jin, and Z. Xie, "High performance offline handwritten chinese character recognition using googlenet and directional feature maps," in *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*. IEEE, 2015, pp. 846–850.

[15] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[16] X.-Y. Zhang, Y. Bengio, and C.-L. Liu, "Online and offline handwritten chinese character recognition: A comprehensive study and new benchmark," *Pattern Recognition*, vol. 61, pp. 348–360, 2017.

[17] C.-L. Liu, "Normalization-cooperated gradient feature extraction for handwritten character recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 8, pp. 1465–1469, aug 2007.

[18] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[20] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European conference on computer vision*. Springer, 2014, pp. 818–833.

[21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *Computing Research Repository (CoRR)*, vol. abs/1409.1556, 2014.

[22] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *CoRR*, vol. abs/1502.03167, 2015.

[23] A. K. Jain, *Fundamentals of Digital Image Processing*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1989.

[24] K. Crammer and Y. Singer, "On the algorithmic implementation of multiclass kernel-based vector machines," *Journal of machine learning research*, vol. 2, no. Dec, pp. 265–292, 2001.

[25] C.-L. Liu, F. Yin, D.-H. Wang, and Q.-F. Wang, "Casia online and offline chinese handwriting databases," in *2011 International Conference on Document Analysis and Recognition*. IEEE, 2011, pp. 37–41.

[26] M. D. Zeiler, "Adadelta: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.