

Design of a Very Compact CNN Classifier for Online Handwritten Chinese Character Recognition Using DropWeight and Global Pooling

Xuefeng Xiao, Yafeng Yang, Tasweer Ahmad, Lianwen Jin* and Tianhai Chang

School of Electronic and Information Engineering, South China University of Technology, Guangzhou, China
xiaoxuefengchina, yangyafeng17, tasveerahmad, lianwen.jin@gmail.com

Abstract—Currently, owing to the ubiquity of mobile devices, online handwritten Chinese character recognition (HCCR) has become one of the suitable choice for feeding input to cell phones and tablet devices. Over the past few years, larger and deeper convolutional neural networks (CNNs) have extensively been employed for improving character recognition performance. However, its substantial storage requirement is a significant obstacle in deploying such networks into portable electronic devices. To circumvent this problem, we propose a novel technique called DropWeight for pruning redundant connections in the CNN architecture. It is revealed that the proposed method not only treats streamlined architectures such as AlexNet and VGGNet well but also exhibits remarkable performance for deep residual network and inception network. We also demonstrate that global pooling is a better choice for building very compact online HCCR systems. Experiments were performed on the ICDAR-2013 online HCCR competition dataset using our proposed network, and it is found that the proposed approach requires only 0.57 MB for storage, whereas state-of-the-art CNN-based methods require up to 135 MB; meanwhile the performance is decreased only by 0.91%.

Keywords—Convolutional neural network, Online handwritten Chinese character recognition, CNN Compression

I. INTRODUCTION

Over the last five decades [1], [2], handwritten Chinese character recognition (HCCR) has attracted considerable attention from researchers and has extensively been studied owing to the large number of character classes, similarity between characters, and variation in writing style. Handwriting recognition can broadly be categorized into online and offline handwriting recognition. The main motivation of this paper is to deal with the problem of storage capacity for online HCCR by using some novel techniques such as GoogleNet. In contrast to offline HCCR, in which gray-scale images are analyzed and classified into different groups, for online HCCR, pen trajectories are the main source of information to recognize different characters [3]. Moreover, online HCCR finds numerous applications in pen input devices, personal digital assistants, smart phones, touch-screen devices, etc.

Academic and commercial research in HCCR has greatly progressed owing to handwriting recognition competitions held over the past few years [4]–[6]. In these competitions, many participants began to use methods based on convolu-

tional neural networks (CNNs) for HCCR, instead of conventional machine learning tools such as the MQDF-classifier [1]. It was also demonstrated that methods based on CNNs can learn more discriminative representations from raw data and can lead to end-to-end solutions for HCCR problems. For the ICDAR 2013 online HCCR competition dataset [6], the novel DropSample training method was proposed in [7], which achieved an accuracy of 97.23% and subsequently of 97.51% when ensembling nine model. Zhang et al. [8] combined conventional normalization-cooperated direction-decomposed feature maps and CNNs to achieve an accuracy of 97.55% and of 97.64% by voting on three models. Thus far, the state-of-the-art CNN-based architecture has produced an accuracy of 97.79% [9] by using the DropDistortion training strategy.

Currently, CNN-based methods are quite popular to deal with the problems of character recognition, and it seems intuitive that the deployment of CNN in portable devices would improve the performance of online handwriting recognition. However, they demand a considerable amount of storage and memory bandwidth. For online HCCR, the aforementioned state-of-the-art methods [7]–[9] require storage spaces of 135.0 MB, 70.50 MB, and 19.03 MB, respectively. This requirement of large storage space is the main hindrance in deploying such deep networks into portable devices such as mobile phones. The large and deep networks are not a pragmatic choice for on-chip storage as they demand additional memory resources. This problem has led to the proposal of some compact designs that would be viable to deploy in portable electronic gadgets.

Recently, many researchers have attempted to build compact networks. Prominently, network pruning [10], [11] is the most effective method to compress CNNs by pruning the redundant connections in each layer. However, to our knowledge, no study has investigated whether these methods are feasible for large-scale online HCCR involving more than 3,700 classes of characters. In previous [12], they adopted the network pruning technique to compress a model built for offline HCCR to 2.3 MB. In the present paper, we propose the use of the DropWeight technique for online HCCR. We also reveal that the DropWeight technique is immune to network architectures and that it can be applied to compress various types of deep network structures, such as VGGNet [13], GoogLeNet [14],

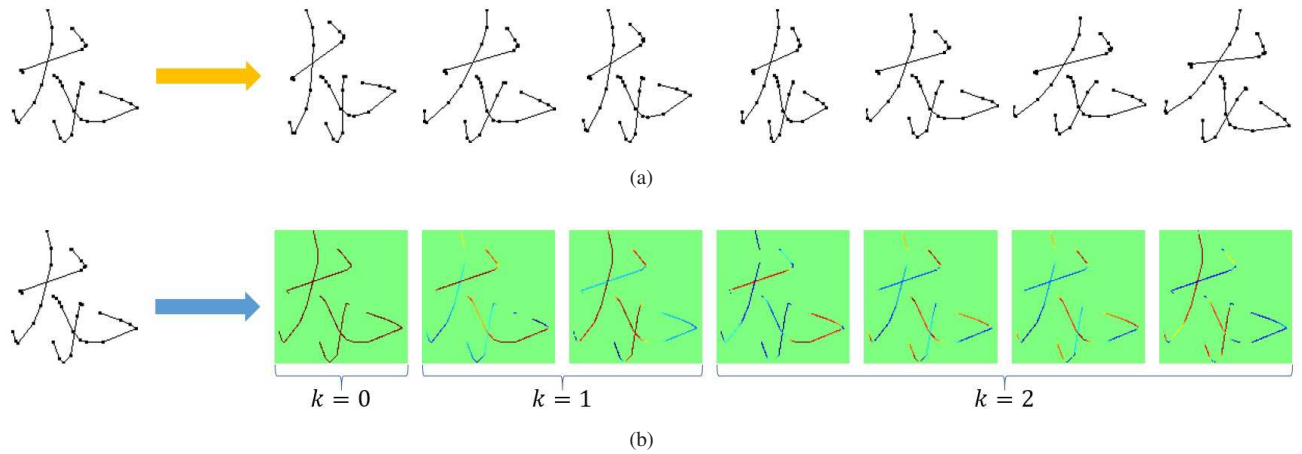


Fig. 1. (a) Characters distortion. (b) Path signature feature map of input pen-tip trajectories.

and ResNet [15]. In the present work, we also demonstrate that the use of global pooling is a good choice for building compact online HCCR systems. We conducted an experiment with the ICDAR 2013 online HCCR competition dataset, based on which we carefully designed deeper and thinner networks that use global pooling, which only costs 9.9 MB storage before connections pruning. After integrating our DropWeight technique, the model costs only 0.57 MB storage. The accuracy of this model is slightly lower than that of the state-of-the-art CNN-based network by 0.91%, but it costs only 1/33 of the storage required for the best CNN model so far reported in the literature.

The remainder of this paper is organized as follows. Section II gives a brief description of the DropWeight technique. Section III highlights various contemporary network architectures. Experiments and results are presented in section IV, and section V concludes the results.

II. DROPWEIGHT

As CNN architectures are very large in size, it is quite desirable to compress the networks by pruning redundant connections. Therefore, the network connections of a network trained in advance are pruned with the idea that weights lower than a threshold should be removed, thereby converting a dense, fully-connected network to a sparse network [10]. The network is pruned and retrained iteratively so that its performance does not degrade significantly.

As proposed by [10], [11], the fixed pruning threshold is computed as follows:

$$P_{th} = \frac{\eta}{N} \sum_{i=1}^N |w_i| + \beta \sqrt{\frac{1}{N} \sum_{i=1}^N (w_i - \frac{1}{N} \sum_{i=1}^N w_i)^2} + \lambda. \quad (1)$$

For the layer containing N weights, the pruning threshold P_{th} is mainly dependent on the average absolute value and variance of weights of layer w_i , but the value of this threshold can also be empirically determined by varying the parameters η, β and λ . However, the application of a fixed threshold is

complicated, as an excessively high threshold would remove many significant connections at the start, and it will be very difficult for the network to recover the original performance; conversely, if the threshold is too low, the desired compression ratio may not be achieved.

To address this problem, we adopt the DropWeight technique, in which the threshold is gradually increased. In experiments, we prune the connections after every I iterations (in experiments, we set $I = 10$). If we wish to prune a certain percentage of connections for a layer, the pruning number must be increased for each pruning iteration. Therefore, the threshold is determined by the pruning number. The absolute values of weights below this threshold are set to zero. By dynamically increasing the pruning number, the threshold would also be gradually increased. During iterations without the pruning process, the weights are updated with a gradient, and the pruned weights cannot be retrieved. Once the desired pruning ratio is reached, the increasing threshold is fixed and noted for further pruning of the layer until pruning ends. Finally, the weight quantization topology, as proposed by [16], is incorporated, following which this quantized-pruned network is fine-tuned for improving performance.

III. NETWORK ARCHITECTURES FOR ONLINE HCCR

A. Characters Distortion

A potential problem in online character recognition is the variation in handwriting style. To address this problem, the concept of character distortion is introduced to generate a large number of training samples artificially. Character distortion is produced by introducing an affine transformation and its variants to the training samples [17]. In character distortion, a nonlinear normalization, as proposed by [18], is also entertained for character shearing and stroke distortion.

Let α be the total degree of character distortion, θ a number ranging from $(-\alpha, \alpha)$, $[x, y]$ the pixel coordinates before transformation, and $[x', y']$ the pixel coordinates after transformation. Then, affine transformations are formulated as follows:

$$[x', y'] \Leftarrow [x, y] \cdot \begin{bmatrix} 1 + \alpha_x & 0 \\ 0 & 1 + \alpha_y \end{bmatrix}, \quad (2)$$

$$[x', y'] \Leftarrow [x, y] \cdot \begin{bmatrix} 1 & \alpha \\ 0 & 1 \end{bmatrix}, \quad (3)$$

$$[x', y'] \Leftarrow [x, y] \cdot \begin{bmatrix} 1 & 0 \\ \alpha & 1 \end{bmatrix}, \quad (4)$$

$$[x', y'] \Leftarrow [x, y] \cdot \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix}, \quad (5)$$

Eq.2 and Eq.3 tilts the strokes; Eq.4 stretches or shrinks strokes, and Eq.5 generates rotational distortion.

As shown in Fig.1(a), global character stretching, scaling, rotation, and translation are performed using affine transformations, whereas local distortion is performed using one-dimensional deformation and non-linear normalization, as proposed by [18]. Non-linear normalization produces character shearing and stroke distortion.

B. Path Signatures

The idea of path signatures was originally proposed by Chen et al. [19] as an iterated integral for solving differential equations. This concept of path signatures was implemented as a set of features by [20] to improve the performance of CNNs for online handwritten character recognition. Empirically, it is revealed that the first and second iterated integrals entail significant information for CNNs. Mathematically, for positive integers k and intervals $[s, t]$, the k -th iterated integral of X is the d^k -dimensional vector defined by

$$I_{s,t}^k = \int_{s < u_1 < \dots < u_k < t} 1 dX_{u_1} \otimes \dots \otimes dX_{u_k}, \quad (6)$$

where \otimes denotes the tensor product. For $k=0$, the iterated integral is 1 and corresponds to its offline image; for $k=1$, the iterated integral corresponds to path displacement; and for $k=2$, the iterated integral corresponds to path curvature, as shown in Fig.1(b).

C. Network Structure

For online HCCR, we designed three different network structures. The first is a conventional streamlined CNN, as shown in Fig.2(a). In this network, all convolutional filters were 3×3 size with one padding pixel for retaining the original size. A max-pooling operation was performed over a 3×3 window with a stride of 2. All convolutional layers and the first fully connected layer were equipped with a batch normalization (BN) [21] layer, and PReLU [22] was added to each BN layer. The overall architecture can be represented as Input-128C3-MP3-160C3-160C3-MP3-256C3-256C3-MP3-384C3-384C3-MP3-1024FC-Output. Then, to reduce the original network's storage size, we use a global average pooling (GAP) layer to replace the last pooling layer and the first fully connected layer. We refer to the two networks as HCCR-Str-FC and HCCR-Str-GAP, respectively.

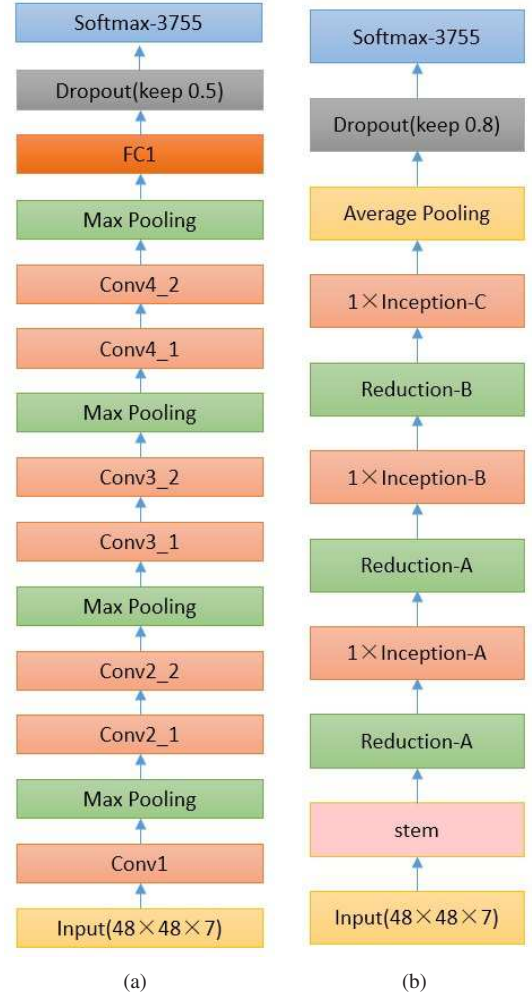


Fig. 2. Network structure of (a) HCCR-Str-FC and (b) HCCR-Inc-GAP.

The second network structure we used is residual network [15], which introduces the short-cut connections to smoothly pass the gradients into shallow layers for solving the problem of vanishing gradients. This architecture won the first place in the ILSVRC 2015 classification challenge. We used an 18-layer architecture in which the output channels of all convolution layers were decreased by 50%. For comparison, we used a fully connected layer that contains 1024 neurons to replace the original global pooling layer. We refer to the two networks as HCCR-Res-FC and HCCR-Res-GAP, respectively.

The last network structure we used is the Inception-v4 [23] network. It is mainly based on GoogLeNet or Inception-v1 [14], which was introduced to address the challenges of memory utilization and computational cost. In order to make it suitable for online HCCR, we removed the first two convolutional layers in the stem module and added a Reduction-A block between the stem block and Inception-A block. As shown in Fig.2(b), we only used three inception modules, and the output channels of all convolution layers were decreased in size by 75%. For fair comparison, we used a fully connected layer that contains 1024 neurons to replace

the original global pooling layer. We refer to the two networks as HCCR-Inc-FC and HCCR-Inc- GAP, respectively.

IV. EXPERIMENT

A. Network Training

Our proposed framework was evaluated using online HCCR dataset. The network was trained using the OLHWDB1.0 and OLHWDB1.1 datasets [24], and the performance of the proposed network was tested using the On-ICDAR2013 dataset [6], which contained 3,755 classes. The network was trained using a set of 2,693,183 samples from 720 different subjects, whereas it was evaluated using 224,590 test images from 60 different writers.

For training the online HCCR network, the distortion technique was used for each sample at each training epoch. Path signature feature maps were extracted from online handwritten Chinese characters, and these feature maps were fed as the input for training the network. The baseline model was trained on the Caffe [25] deep learning platform with a mini-batch size of 128 and momentum of 0.9. The learning rate was initialized with 0.1, and it was reduced by 0.1 after every 70,000 iterations. The training process concluded after 300,000 iterations.

TABLE I
COMPRESSION RESULTS FOR DIFFERENT NETWORK STRUCTURES

Model	Before Compression		After Compression	
	Stor.(MB)	Accu.(%)	Stor.(MB)	Accu.(%)
HCCR-Str-GAP	19.24	97.51	0.84	96.62
HCCR-Str-FC	41.93	97.77	1.18	96.49
HCCR-Res-GAP	14.40	96.89	0.70	96.05
HCCR-Res-FC	29.42	97.02	1.07	96.03
HCCR-Inc-GAP	9.36	97.45	0.57	96.88
HCCR-Inc-FC	56.05	97.65	0.76	96.83

B. Accuracy and Storage

Tab.I presents a comparative analysis of storage and accuracy for the ICDAR-2013 online competition database for our six proposed networks. For the same structure, we can find that the use of global pooling to replace the fully connected layer slightly decreases the performance but significantly decreases the storage space required. Therefore, by using the DropWeight technique for the six networks, the storage capacity is drastically decreased, whereas the accuracy is only slightly decreased. For the streamlined, residual, and inception-based network, it is initially observed that the global pooling layer achieves a slightly lower accuracy compared to that of the fully connected layer. However, after compression, the performance of global pooling networks is better than those of fully connected layer networks; in addition, the storage required for the former is lower than that required for the latter. Thus, it is clearly demonstrated that global pooling is a good choice to build a compact system for online HCCR.

Tab.II illustrates the results of three previous CNN-based methods [4]–[6] that have achieved the highest performance thus far on the ICDAR-2013 online database. It is clear that

our HCCR-Inc-GP can achieve a very compact design as compared with the three previous architectures, costing only 9.9 MB of memory. Moreover, after further compression, it costs merely 0.57 MB of memory and can still reach an accuracy of 96.88%, which is certainly higher but requires 210 times smaller storage compared to the conventional method (DFE + DLQDF) [26]. Even compared with the state-of-the-art CNN models for online HCCR, our model is 1/33 times more cost efficient while the performance is decreased only by 0.91%.

TABLE II
RESULT FOR ICDAR-2013 ONLINE HCCR COMPETITION DATASET

Method	Ref.	Storage(MB)	Accuracy(%)
Traditional Method: DFE+DLQDF	[26]	120.0	95.31
DropSample	[7]	135.0	97.51
DirectMap+ConvNet	[8]	70.50	97.64
DropDistortion	[9]	19.03	97.79
HCCR-Inc-GAP	ours	9.90	97.45
HCCR-Inc-GAP-Pruned	ours	0.57	96.88

V. CONCLUSION

In this paper, we proposed the DropWeight technique to compress popular CNN architectures for online HCCR, which includes a streamlined, residual, and inception-based network. We also demonstrated that global pooling is a good choice to build a compact network for online HCCR. Finally, we built a network that costs only 0.57 MB of storage but can still achieve an accuracy comparable to those of state-of-the-art CNN models. In the future, we will extend the method to other deep learning model such as long short-term memory (LSTM) network to address the problem of online handwritten text recognition and natural language processing.

REFERENCES

- [1] F. Kimura, K. Takashina, S. Tsuruoka, and Y. Miyake, “Modified quadratic discriminant functions and the application to chinese character recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9, no. 1, pp. 149–153, 1987.
- [2] R. Dai, C. Liu, and B. Xiao, “Chinese character recognition: history, status and prospects,” *Frontiers of Computer Science in China*, vol. 1, no. 2, pp. 126–136, 2007.
- [3] C. Liu, S. Jäger, and M. Nakagawa, “Online recognition of chinese characters: The state-of-the-art,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 198–213, 2004.
- [4] C.-L. Liu, F. Yin, D.-H. Wang, and Q.-F. Wang, “Chinese handwriting recognition contest 2010,” in *Proceedings of Chinese Conference on Pattern Recognition (CCRP)*. IEEE, 2010, pp. 1–5.
- [5] C.-L. Liu, F. Yin, Q.-F. Wang, and D.-H. Wang, “Icdar 2011 chinese handwriting recognition competition,” in *Proceedings of International Conference on Document Analysis and Recognition (ICDAR)*. IEEE, 2011, pp. 1464–1469.
- [6] F. Yin, Q. Wang, X. Zhang, and C. Liu, “ICDAR 2013 chinese handwriting recognition competition,” in *Proceedings of International Conference on Document Analysis and Recognition (ICDAR)*, 2013, pp. 1464–1470.
- [7] W. Yang, L. Jin, D. Tao, Z. Xie, and Z. Feng, “DropSample: A new training method to enhance deep convolutional neural networks for large-scale unconstrained handwritten chinese character recognition,” *Pattern Recognition*, vol. 58, pp. 190–203, 2016.
- [8] X. Zhang, Y. Bengio, and C. Liu, “Online and offline handwritten chinese character recognition: A comprehensive study and new benchmark,” *Pattern Recognition*, vol. 61, pp. 348–360, 2017.

- [9] S. Lai, L. Jin, and W. Yang, "Toward high-performance online HCCR: a CNN approach with dropdistortion, path signature and spatial stochastic max-pooling," *Pattern Recognition Letters*, vol. 89, pp. 60–66, 2017.
- [10] S. Han, J. Pool, J. Tran, and W. J. Dally, "Learning both weights and connections for efficient neural network," in *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 2015, pp. 1135–1143.
- [11] Y. Guo, A. Yao, and Y. Chen, "Dynamic network surgery for efficient dnns," in *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 2016, pp. 1379–1387.
- [12] X. Xiao, L. Jin, Y. Yang, W. Yang, J. Sun, and T. Chang, "Building fast and compact convolutional neural networks for offline handwritten chinese character recognition," *CoRR*, vol. abs/1702.07975, 2017.
- [13] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proceedings of International Conference on Learning Representations (ICLR)*, 2014.
- [14] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9.
- [15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.
- [16] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural network with pruning, trained quantization and huffman coding," in *Proceedings of International Conference on Learning Representations (ICLR)*, 2016.
- [17] W. Yang, L. Jin, Z. Xie, and Z. Feng, "Improved deep convolutional neural network for online handwritten chinese character recognition using domain-specific knowledge," in *Proceedings of International Conference on Document Analysis and Recognition (ICDAR)*, 2015, pp. 551–555.
- [18] K. Leung and C. H. Leung, "Recognition of handwritten chinese characters by combining regularization, fisher's discriminant and distorted sample generation," in *Proceedings of International Conference on Document Analysis and Recognition (ICDAR)*, 2009, pp. 1026–1030.
- [19] K.-T. Chen, "Integration of paths—a faithful representation of paths by noncommutative formal power series," *Transactions of the American Mathematical Society*, vol. 89, no. 2, pp. 395–407, 1958.
- [20] B. Graham, "Sparse arrays of signatures for online character recognition," *CoRR*, vol. abs/1308.0371, 2013.
- [21] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of International Conference on Machine Learning (ICML)*, 2015, pp. 448–456.
- [22] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of International Conference on Computer Vision (ICCV)*, 2015, pp. 1026–1034.
- [23] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proceedings of Association for the Advancement of Artificial Intelligence (AAAI)*, 2017, pp. 4278–4284.
- [24] C. Liu, F. Yin, D. Wang, and Q. Wang, "CASIA online and offline chinese handwriting databases," in *Proceedings of International Conference on Document Analysis and Recognition (ICDAR)*, 2011, pp. 37–41.
- [25] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. B. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of International Conference on Multimedia (ICM)*, 2014, pp. 675–678.
- [26] C. Liu, F. Yin, D. Wang, and Q. Wang, "Online and offline handwritten chinese character recognition: Benchmarking on new databases," *Pattern Recognition*, vol. 46, no. 1, pp. 155–162, 2013.