2016 23rd International Conference on Pattern Recognition (ICPR)
Cancún Center, Cancún, México, December 4-8, 2016

Handwritten Chinese Character Recognition with Spatial Transformer and Deep Residual Networks

Zhao Zhong*‡, Xu-Yao Zhang*, Fei Yin*, Cheng-Lin Liu*†‡
*National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences,
95 Zhongguan East Road, Beijing 100190, P.R. China
†CAS Center for Excellence in Brain Science and Intelligence Technology,
95 Zhongguan East Road, Beijing 100190, P.R. China
‡University of Chinese Academy of Sciences, Beijing, P.R. China
Email: {zhao.zhong, xyz, fyin, liucl}@nlpr.ia.ac.cn

***Abstract*—This paper considers using deep neural networks for handwritten Chinese character recognition (HCCR) with arbitrary position, scale, and orientations. To solve this problem, we combine the recently proposed spatial transformer network (STN) with the deep residual network (DRN). The STN acts like a character shape normalization procedure. Different from the traditional heuristic shape normalization methods, STN is learned directly from the data. Furthermore, the DRN makes the training of very deep network to be both efficient and effective. With the combination of STN and DRN, the whole model can be trained jointly in an end-to-end manner. In this paper, new state-of-the-art performance has been achieved by our proposed model on the offline ICDAR-2013 Chinese handwriting competition database. Moreover, the experiment on randomly distorted samples shows that the STN is very effective for robust HCCR in rectifying the shape of distorted characters.**

I. Introduction

Handwritten Chinese character recognition (HCCR) is a challenging task which has been studied for more than fifty years, due to the large number of character classes and the distinct handwriting styles across individuals. In traditional methods, the procedures for HCCR usually include: shape normalization, feature extraction, dimensionality reduction, and classifier training. For example, the MQDF and DLQDF [1], [2] based approaches have achieved good performance in past evaluations [3], however, the traditional methods are reaching their bottlenecks nowadays [4], [5].

Due to the record-breaking performance, deep neural network, especially convolutional neural network (CNN) [6], is becoming more and more popular in recent years. Many CNN-based models have shown state-of-the-art performance in classification [7], [8], [9], localization [10], semantic segmentation [11] tasks, and so on. Meanwhile, in HCCR domain, various CNN-based models have been proposed, and the records on HCCR have been renewed many times [12], [13], [14], [5].

The first reported successful application of CNN for large vocabulary HCCR was the multi-column deep neural network (MCDNN) proposed by Ciresan et al. [12]. After that, the team from Fujitsu used a CNN-based model to win the first place in the ICDAR-2013 HCCR competition [4], with an accuracy rate of 94.77%. Recently, Zhong et al. [14] improved the performance to 96.35% by a single model of HCCR-Gabor-GoogLeNet which outperformed the human performance. Li et al. [15] from Fujitsu further improved the performance to 96.58% based on a single CNN model with augmented training data using distortion. The ensemble based methods can be further used to improve the performance with some tradeoff on speed and memory.

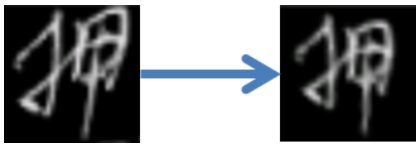

Fig. 1. The spatial transformer network (STN) rectifies the original handwritten Chinese character image (left) to the regular handwritten Chinese characters image (right) . The STN can be seen as a shape normalization preprocess in HCCR framework, and the STN is directly learned from data in a discriminative manner.

Although significant progress has been achieved by using CNN-based models for HCCR, most models ignore the information of spatial distortion for handwritten Chinese characters in real world. Due to different handwriting environment and styles, the direction, position, and size of the character may vary significantly for different samples. Convolutional neural networks are still lacking the ability to be spatially invariant with respect to the input data. Traditionally, the shape normalization methods [16] are used to transform the samples into fixed-size and normalized characters, which are very important for the following classification task. However, these normalization methods are heuristic and not guaranteed to be optimal for the task of HCCR. In light of this, motivated by the spatial transformer network (STN) [17], we propose a new model for HCCR by directly learning the shape normalization form the data in an end-to-end manner. The proposed method is robust to irregular (different size and orientations) characters. To further improve the performance, we also combine the STN with the recently proposed very deep networks, namely, deep residual network (DRN) [18]. With the combination of STN and DRN, we can achieve a new state-of-the-art result on the ICDAR-2013 competition database. Moreover, we also show that STN is important and effective for the recognition of

978-1-5090-4846-5/16/$31.00 ©2016 IEEE

distorted characters.

Specifically, the STN [17] is used to transform the input character image (with arbitrary size or orientation) into a rectified image including scaling, cropping, and rotation. As shown in Fig. 1, with STN, the character shape is normalized compared with the original image. The transformation used in STN is a 2D affine transformation configured by 6 transformation parameters, which are directly learned from the input image. Due to the shape normalization ability of STN, the subsequent classifier (convolutional neural network) can deal with the distorted handwritten Chinese characters properly and lead to superior classification performance. After STN, we apply the DRN [18] as the final classification model. DRN is a powerful and super deep framework which has been used to win the ILSVRC [19] and COCO [20] 2015 competitions in classification, detection, localization and segmentation recently. Due to its special structure, DRN is much easier for optimization with the BP algorithm compared with other deep models. Moreover, due to the considerably increased depth, the classification accuracy can also be significantly improved.

The whole model is trained in an end-to-end manner by jointly optimizing the STN and DRN models on the raw data. The STN can learn the transformation parameters automatically without any extra supervision, e.g, the geometric or position information. The parameters in STN are learned with respect to the backward gradients from the DRN part. In our experiments, we find that the STN tends to produce frontal and horizontal character which can simplify the subsequent classification task. The main contribution of this paper is a robust HCCR system based on deep neural network model, whose network architecture is specifically designed for handwritten Chinese characters in the real world. Two experiments are conducted to show that the proposed STN+DRN model can not only achieve state-of-the-art performance in traditional HCCR task but also being robust to irregular characters.

The rest of this paper is organized as follows. Section II introduces the proposed HCCR framework including STN, DRN and the whole system. Section III presents the experimental results and Section IV concludes this paper.

II. The Proposed HCCR Framework

The architecture of our proposed HCCR framework includes two parts: spatial transformer network (STN) and deep residual network (DRN), and we will describe each part in detail in the following subsections.

A. Spatial Transformer Network

As shown in Fig. 2, the STN [17] includes three parts. First, a localization network takes the input image, and through a number of hidden layers, it regresses the parameters of the spatial transformation that should be applied to the original image. Then, the predicted transformation parameters are used to create a sampling grid, i.e., a set of points where the input image should be sampled to produce the transformed output, which is implemented by the sampling grid generator. Finally, the sampler takes the sampling grid and the input image to produce the output image, which is sampled from the input at the grid points. The combination of these three components forms the STN. An advantage of STN is that all components are differentiable, which means the STN can back-propagate loss-gradient in the neural network, and hence, STN can be combined and jointly trained with other networks.

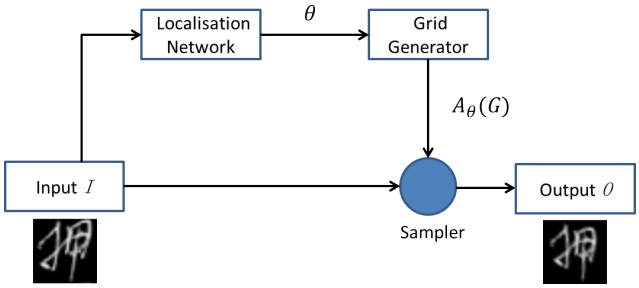

Fig. 2. The architecture of spatial transformer network (STN) [17]. The input image I is passed to a localization network to produce parameters θ. The grid generator then produces the sampling grid $A_\theta\,(G)$. At last, both the I and the $A_\theta\,(G)$ are used by the sampler to produce the rectified output image O.

1) Localization Network: The localization network takes the input image $I \in \Re^{H\times W\times C}$ (height H, width W, channels C) to output θ, the 6-dimensional parameters of the affine transformation A_θ to be applied on the original image:

$$\theta = f_{loc}(I) \tag{1}$$

The localization network function $f_{loc}()$ can take any form, such as a fully-connected network or a convolutional network. In our work, we use the localization network that contains two convolution+maxpooling layers, followed by two fully-connected layers for the regression of the transformation parameters θ. The CNN based localization network is trained jointly with the following networks without the need of any extra geometric or position supervision during training.

2) Grid Generator: The grid generator generates a sampling grid G for the "Sampler" to perform a warping of the input image I [17]. The output pixels are defined to lie on a regular grid $G = \{G_i\}$ of pixels $G_i = (x_i^t, y_i^t)$, forming an output image $O \in \Re^{H'\times W'\times C}$, where H' and W' are the height and width of the grid, and C is the number of channels which is the same as the input image. In our case, the pointwise affine transformation is

$$\begin{pmatrix} x_i^s \\ y_i^s \end{pmatrix} = A_\theta \begin{pmatrix} x_i^t \\ y_i^t \\ 1 \end{pmatrix} = \begin{bmatrix} \theta_{11} & \theta_{12} & \theta_{13} \\ \theta_{21} & \theta_{22} & \theta_{23} \end{bmatrix} \begin{pmatrix} x_i^t \\ y_i^t \\ 1 \end{pmatrix} \tag{2}$$

where (x_i^t, y_i^t) is the target coordinate of the regular grid in the output image, (x_i^s, y_i^s) is the source coordinate in the input image that define the sample points, and A_θ is the affine transformation matrix. All coordinates are normalised in $[-1, +1]$. In practice, we initialize the affine transformation matrix A_θ as follows:

$$A_\theta = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{3}$$

The transform defined in Eq. (2) allows rotation, scale, cropping, translation and skew to be applied to the input image, and requires only 6 affine transformation parameters from the localization network. The grid generation involves only matrix production. Therefore, the grid generator is differentiable. As illustrated in Fig. 2, applying the parameterised sampling grid to an input image will produce regular output image.

3) Sampler: Lastly, the sampler take the set of sampling points $A_\theta(G)$ along with the input image I, producing the sampled output image O. We use a bi-linear sampler S to interpolate the pixels (x_i^t, y_i^t) in the pixels near (x_i^s, y_i^s) on the input image. Setting all the pixels in O, we get the rectified image:

$$O = S(I, A_\theta(G)) \tag{4}$$

The bi-linear sampler S is also a differentiable function, which means that all STN components can be trained with normal gradient descent methods. The affine transformation allows the STN to transform irregular images into rectified regular characters. As shown in the later sections and also Fig. 5, the irregular characters (include skew, different scale, and rotated characters) can be effectively normalized by the STN.

B. Deep Residual Network

Large vocabulary HCCR is a challenging task that requires powerful models like DNN, which naturally integrates low/mid/high level features [21] and classifiers in an end-to-end framework. However, when the network becomes deeper, it will encounter the degradation problem as reported in [22]. To solve this problem, He et al. [18] proposed an effective framework: deep residual network (DRN).

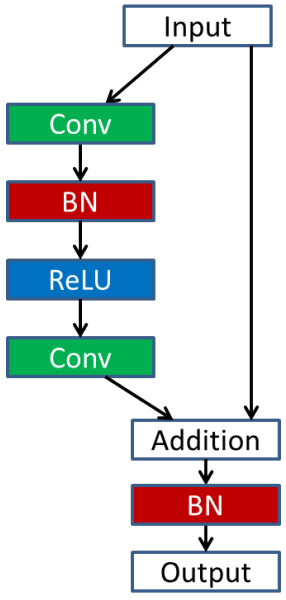

Fig. 3. The building block architecture in deep residual network: the standard convolution layer (Conv), batch normalization (BN), and ReLU. We remove the ReLU activation and add the BN after the addition.

1) Residual Learning: It is hypothesized [18] that optimizing the residual mapping is much easier than the original mapping. The residual mapping is defined as:

$$F(x) = H(x) - x \tag{5}$$

$H(x)$ denotes the desired mapping, x denotes the input feature map, and the original mapping is recast into $F(x) + x$.

2) Shortcuts: The formulation of $F(x) + x$ can realized by shortcut connections [23] in neural network. Shortcut connections are those skipping one or more layers. We adopt residual learning to every few stacked layers, and the definition can be expressed as:

$$y = F(x, \{W_i\}) + x \tag{6}$$

Here x and y are the input and output feature map of the layers. The function $F(x, \{W_i\})$ represents the residual mapping to be learned. With this framework, much more neural network layers can be stacked to produce deeper networks for the purpose of higher classification performance but still having lower complexity and less parameters compared with traditional deep models.

In our work, the residual building block is defined as Fig. 3, which is a little different from [18]. We use a stack of two convolutional layers followed by batch normalization layers [24]. The shortcut connection performs as an identity mapping in the building block, and we add the shortcut connection to the convolutional layers. The ReLU activation is removed and the BN is added after the addition. This architecture can be trained easier and its convergence is faster. Since the residual building block is differentiable, the whole model can be trained end-to-end.

C. The Whole HCCR Framework

As shown in Fig. 4, by combining STN and DRN, we build a deep convolutional neural network for HCCR. At the bottom of the network, the STN module transforms the input image to rectified image. Then, the following DRN module is used to classify each rectified image and outputs the posterior probabilities. The whole system is trained end-to-end with the raw data and the ground truth of the character.

In the localization network part, we use the network architecture as: max[2,2]-conv[20,5,1,0]-max[2,2]-conv[20,5,1,0]-fc[20]-fc[6]. The "max", "conv", "fc" denote the max-pooling layer, convolutional layer and fully-connected layer respectively. The number in the bracket for max means [filter size, stride], for conv means [number of units, filter size, stride, padding size] and for fc means [number of units]. After the convolutional layers and fully-connected layers, we use ReLU [26] as the activation function.

In this paper, we consider two residual networks: a 19-layer and a 34-layer residual networks. The number of units in the softmax layer is 3755 (corresponding to the number of classes). Fig. 4 shows the architecture for residual-19 network.[1] Like the standard residual network, our 19-layer

[1]To save space, we omit the architecture for the residual-34 network.

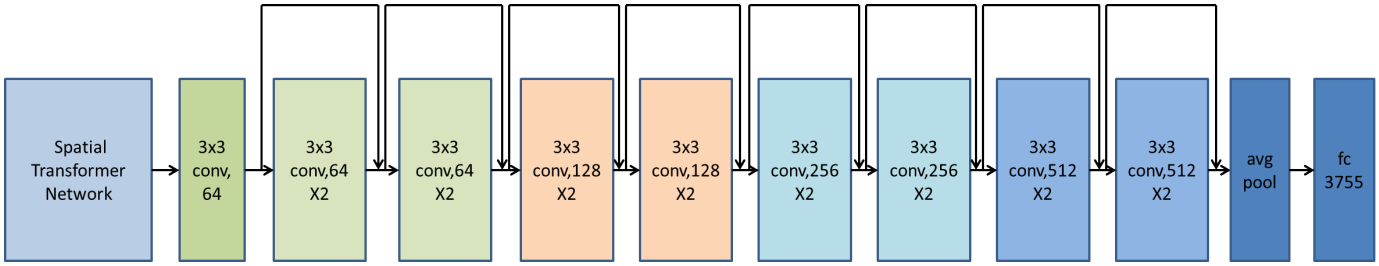

Fig. 4. The architecture of the proposed HCCR network which includes two parts: 1) spatial transformer network which rectifies the input handwritten Chinese character image; 2) deep residual network which predicts the label distribution for rectified character image. The whole model is trained end-to-end in an unified framework.

TABLE I
RESULTS ON ICDAR-2013 OFFLINE HCCR COMPETITION DATABASE

Method	Accuracy	Training Data	Ensemble	Memory
Residual-19	96.36%	HWDB1.0+1.1	no	51.7MB
STN-Residual-19	96.50%	HWDB1.0+1.1	no	51.8MB
Residual-34	**97.36%**	HWDB1.0+1.1	no	92.2MB
STN-Residual-34	**97.37%**	HWDB1.0+1.1	no	92.3MB
Human Performance [4]	96.13%	n/a	n/a	n/a
Traditional Method:DFE+DLQDF [2]	92.72%	HWDB1.0+1.1	no	120.0MB
ICDAR-2011 Winner [25]	92.18%	HWDB1.1	no	27.35MB
ICDAR-2013 Winner [4]	94.77%	HWDB1.1	yes(4)	2.402GB
MCDNN [12]	95.79%	HWDB1.1	yes(8)	349.0MB
ATR-CNN Voting [13]	96.06%	HWDB1.1	yes(4)	206.5MB
HCCR-Gabor-GoogLeNet [14]	96.35%	HWDB1.0+1.1	no	27.77MB
HCCR-Gabor-GoogLeNet-Ensemble [14]	96.74%	HWDB1.0+1.1	yes(10)	270.0MB
CNN-Single [15]	96.58%	HWDB1.0+1.1+1.2	no	190.0MB
CNN-Votiong [15]	96.79%	HWDB1.0+1.1+1.2	yes(5)	950.0MB
DirectMap-ConvNet [5]	96.95%	HWDB1.0+1.1	no	23.5MB

network contain 8 shortcuts building block. We also apply dropout after each shortcuts block for better performance.

In the optimization process, we use a weight decay of 0.0001 and momentum of 0.9, and adopt the weight initialization method of [27], BN [24], and dropout between 0.05-0.1 in a few layers. The models are trained with a mini-batch size of 256 on one GPU. For the STN-Residual-19 we start with a learning rate of 0.01, and for the STN-Residual-34 we start with a learning rate of 0.001 for the STN and 0.01 for the rest part. The learning rates for all models are divided by 10 when both the training loss and the training error stop reducing. We use a very simple data distortion for training, i.e., random rotation between -3° and +3°. For testing, we only evaluate the original 32x32 input image.

III. EXPERIMENTAL RESULTS

A. Experimental Data

The database we used for evaluation is the offline CASIA-HWDB [3] collected by the Institute of Automation of the Chinese Academy of Sciences. We use CASIA-HWDB1.0 database (420 writers) and CASIA-HWDB1.1 database (300 writers) for training, and the ICDAR-2013 Chinese handwriting recognition competition set (60 writers) for testing. The number of character-class is 3755.

B. Experimental Settings

We shuffle our training data at the beginning of every epoch and normalize the offline character image to a size of 32x32 as the input for the network. Our model is implemented under the Torch7 scientific computing platform [28]. We use the CUDA backend and cuDnn accelerated library in our implementation for high-performance GPU acceleration. Our experiments are carried out on a workstation with one Intel(R) Xeon(R) E5-2680v3 2.50GHz CPU, one NVIDIA TitanX GPU, and the 256GB RAM. The proposed model takes 0.95ms to recognize one handwritten Chinese character, and each model is trained for about 3 days after 25 epochs over the training set.

C. Comparison with State-of-the-art

Our model is firstly evaluated on the ICDAR-2013 offline HCCR competition database. We compare our models with different typical methods using the same dataset. The experimental results are shown in Table I. It is shown that we can achieve new state-of-the-art performance for offline HCCR compared with previous approaches. The convolutional neural

TABLE II
RESULTS ON DISTORTED OFFLINE HCCR DATABASE

Method	Accuracy
FCN	41%
STN-FCN	64%
Residual-19	92.62%
STN-Residual-19	93.66%

network (CNN) based approaches can significantly outperform the traditional approaches such as DLQDF. However, previous CNN based approaches for HCCR are not deep enough (i.e., most models have less than 20 layers). In this paper, by using the deep residual network (DRN) [18], we can further improve the performance significantly, which identify the effectiveness of increasing depth for improving accuracy. Moreover, with the help of STN [17], the performance is further improved, which justify the effectiveness of STN in character shape normalization. The combination of STN and DRN is shown to be a good choice for HCCR. Another finding is that: when the classification network becomes more and more powerful, the effectiveness of STN will be reduced, for example, there is no difference between STN-Residual-34 and Residual-34. This may be caused by two reasons: (1) the STN used in this paper is too simple, i.e., an affine transformation with only six parameters, and (2) the character samples in the competition database do not contain large distortions.

Taking all these together, we can conclude that our model can perform state-of-the-art Chinese character recognition in the general setting. However, a much more difficult task is the recognition of characters with arbitrary position, scale, and rotation angles, which is commonly happened in real applications. Therefore, to prove the effectiveness of our model in dealing with the irregular handwritten Chinese characters, we conduct another experiment on recognizing characters with large distortions.

D. Distorted HCCR Experiment

In order to simulate the handwritten Chinese character in real world without any constraints, we distort the character image from CASIA-HWDB dataset and ICDAR-2013 offline HCCR competition datasets. As shown in Fig. 5, we rotate the handwritten Chinese character image by -45° and +45°, randomly scale the character by a factor of between 0.8 and 1.0, place the pixel in a random location in a 42x42 image, and then resize the image to 32x32. All processes are implemented with random values sampled from the uniform distribution.

For the distorted HCCR experiment, we use the pre-trained STN parameters from a simple classifier: STN followed by 3 fully connected layers (FCN). The pre-trained STN parameters is used to give a better initialization. After that, in the jointly training of STN and Residual-19, for the first 10000 iterations the STN parameters are not updated, and then the whole STN and DRN models are trained jointly. This is to give a better transfer of STN from the pre-trained model.

The results for distorted HCCR are summarized in Table II, we can see that the models with STN outperform the models without STN significantly for both FCN and Residual-19. The contribution (improvement) caused by STN for FCN is much more significant compared with Residual-19. This can be explained by that the deep residual network already have some ability to deal with the distortion in a certain degree than traditional neural networks. To give a better illustration, in Fig. 5, we show the transformed characters by the STN model. It is shown that the STN is very effective for rectifying the irregular handwritten Chinese character and then improving the following classification task.

IV. CONCLUSION

In this paper, we present a new HCCR system based on deep neural networks. Our model is designed to be very deep due to the residual structure. The proposed system is an end-to-end model which needs no extra complex preprocesses or extra domain knowledge in the training process. We use the spatial transformer network (STN) module to transform the input handwritten Chinese character images into regular characters, which is then jointly trained with the deep residual network (DRN). The experimental results show that: (1) our model can achieve a new state-of-the-art performance for HCCR, and (2) the proposed HCCR system is robust to the irregular handwritten Chinese characters. In future, we will continue to improve the proposed model from different aspects, such as using deeper residual models and more powerful transformation function in STN to get better performance. We will also try the multi-task framework to deal with the difficulty of STN training when the model becomes deeper.

ACKNOWLEDGMENTS

This work has been supported in part by the National Basic Research Program of China (973 Program) Grant 2012CB316302,the National Natural Science Foundation of China (NSFC) Grants 61573355 and 61403380, the Strategic Priority Research Program of the Chinese Academy of Sciences (Grants XDA06040102 and XDB02060009).

REFERENCES

[1] C.-L. Liu, H. Sako, and H. Fujisawa, "Discriminative learning quadratic discriminant function for handwriting recognition," *IEEE Trans. Neural Networks*, vol. 15, no. 2, pp. 430–444, 2004.

[2] 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.

[3] ——, "CASIA online and offline Chinese handwriting databases," in *Proc. 11th International Conference on Document Analysis and Recognition*. IEEE, 2011, pp. 37–41.

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

[5] 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.

[6] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.

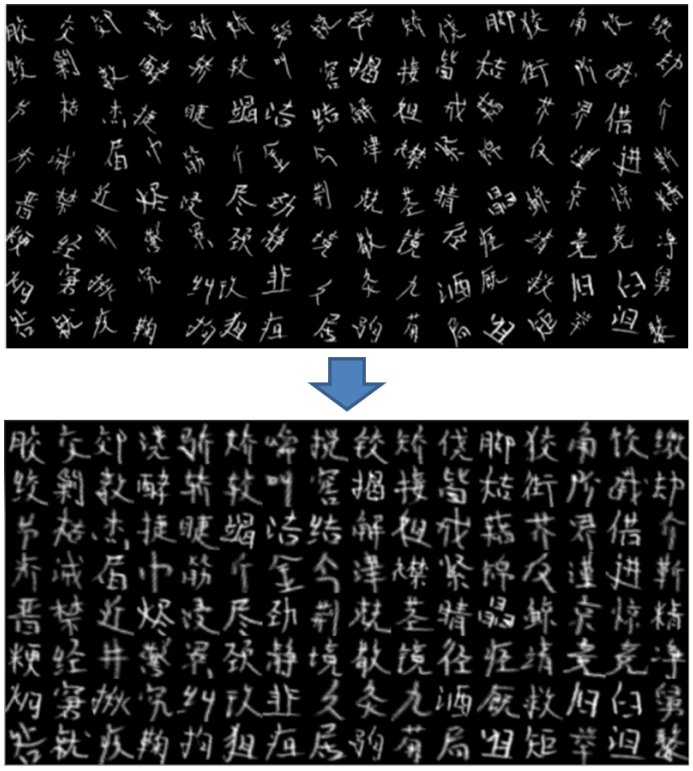

Fig. 5. The effectiveness of the STN in shape normalization for HCCR . Top: the distorted handwritten Chinese characters. Bottom: the rectified characters with the STN transformation.

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[8] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[9] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[10] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler, "Joint training of a convolutional network and a graphical model for human pose estimation," in *Proc. Advances in Neural Information Processing Systems*, 2014, pp. 1799–1807.

[11] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[12] D. Ciresan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 3642–3649.

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

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

[15] L. Chen, S. Wang, W. Fan, J. Sun, and N. Satoshi, "Beyond human recognition: A CNN-Based framework for handwritten character recognition," in *Proc. 3rd IAPR Asian Conference on Pattern Recognition*. IEEE, 2015.

[16] C.-L. Liu and K. Marukawa, "Pseudo two-dimensional shape normalization methods for handwritten Chinese character recognition," *Pattern Recognition*, vol. 38, no. 12, pp. 2242–2255, 2005.

[17] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," in *Proc. Advances in Neural Information Processing Systems*, 2015, pp. 2008–2016.

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

[19] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[20] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proc. 13th European Conference on Computer Vision*. Springer, 2014, pp. 740–755.

[21] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. 13th European Conference on Computer Vision*. Springer, 2014, pp. 818–833.

[22] K. He and J. Sun, "Convolutional neural networks at constrained time cost," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5353–5360.

[23] C. M. Bishop, *Neural networks for pattern recognition*. Oxford university press, 1995.

[24] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd International Conference on Machine Learning*, 2015, pp. 448–456.

[25] C.-L. Liu, F. Yin, Q.-F. Wang, and D.-H. Wang, "ICDAR 2011 Chinese handwriting recognition competition," in *Proc. 11th International Conference on Document Analysis and Recognition*. IEEE, 2011, pp. 1464–1469.

[26] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proc. 27th International Conference on Machine Learning*, 2010, pp. 807–814.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proc. IEEE International Conference on Computer Vision*, 2015, pp. 1026–1034.

[28] R. Collobert, K. Kavukcuoglu, and C. Farabet, "Torch7: A matlab-like environment for machine learning," in *BigLearn, NIPS Workshop*, no. EPFL-CONF-192376, 2011.