Available online at www.ijpe-online.com

International Journal of Performability Engineering

vol. 16, no. 8, August 2020, pp. 1271-1278
DOI: 10.23940/ijpe.20.08.p14.12711278

Improving Font Effect Generation based on Pyramid Style Feature

Feiyu Zhang[a,b], Yi Yang[b], Weixing Huang[b], Guigang Zhang[b], and Jian Wang[b,*]

[a]*School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, 100049, China*
[b]*Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China*

Abstract

The task of font effect generation is stylizing the shape and texture of style images into font images. There exist some methods to handle this task. However, stylized font images become unrecognized when the glyph structure is quite complicated. This paper proposes a font effect generation model based on pyramid style feature. Morphology operations are utilized to improve the transferring effect. Experiments show that our proposed method is more suitable for stylizing complex glyph images than other state-of-the-arts methods.

Keywords: font effect generation; pyramid feature; complex glyph

(Submitted on May 18, 2020; Revised on June 16, 2020; Accepted on July 20, 2020)

1. Introduction

Artistic fonts are creative and special beautification and modification of a traditional font. Artistic font is a new type of decoration with the most representative meaning in font art. The focus is to highlight the inherent meaning of language with the beauty of form. Artistic fonts are widely accepted and become a new artistic creation.

With the rapid development of style transfer, it has brought some inspiration for the task of generating special effects of artistic characters. Researchers train neural network models to generate appealing artist fonts. It greatly saves time and increases efficiency for art designers. However, the current state-of-the-art methods are not very robust. There exist structural disconnection and overlapping strokes when the glyph structure is complicated. These problems directly cause the stylized font images to be unrecognizable.

To solve these problems, we propose a font effect generation model that can maintain glyph structure. As Figure 1 showed, our model first learns the shape features of the style picture and uses the learned structure transformation network to transform the stroke shape of the font picture. Then, it learns the texture characteristics of the style picture and uses the learned texture transformation network to transform the stroke texture of the font picture. Compared to the results generated by the baseline model, our model generated appealing results. Since neural networks are hard to control, we utilized several data preprocessing methods and multi-resolution features. When the glyph structure is sophisticated, there exist different scale strokes and dense strokes. Obviously, it is not enough to use the single-scale features. Multi-scale could help to generate fine texture in dense stroke areas. The generated characters by our model could maintain glyph structure, which allows the stylized character to still be recognized.

In summary, our contributions are as follow:

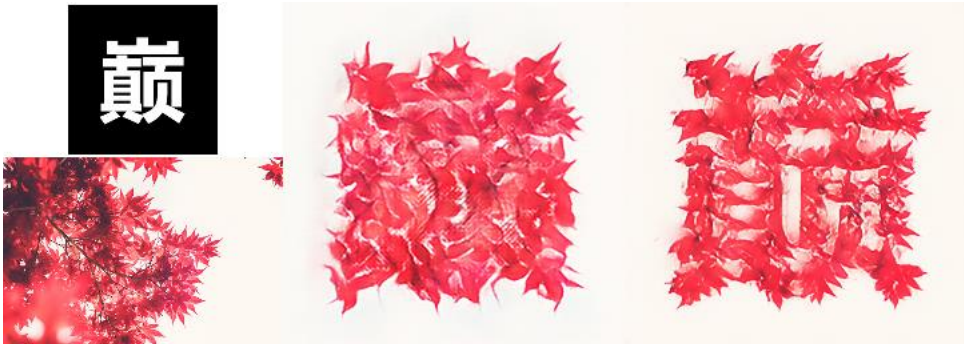
- Spatial pyramid feature. The feature could capture different scale deformations, which ensure that fine grained deformations and coarse grained deformations are both guaranteed to generate.
- Morphology operation. This paper uses corrosion and expansion operation to remove unnecessary connections after binary character images are deformed.

* Corresponding author.
E-mail address: jian.wang@ia.ac.cn

- Contextual loss. This paper utilizes contextual loss to calculate the style distance when images are not strictly aligned. This could reduce the generation of artifacts.

Text image

Style image



Baseline Our method

Figure 1. The image in the top left corner is a Chinese character processed by binarization. The image in the bottom left corner is a style image consisted of maple leaves. The image in the middle is the result transferred by Shape-Matching GAN [1]. The image on the right is the result transferred by our model.

2. Related Work

2.1. Image-to-Image Translation

Image-to-image translation is a class of vision and graphics problems where the goal is to learn the mapping between an input image and an output image. However, designing and learning such models become complicated as there can be arbitrarily large numbers of styles and domains in the dataset.

To address the style diversity, much work on image-to-image translation has been developed. Isola et al. presented an image-to-image translation framework called pix2pix [2]. Pix2pix uses a conditional generative adversarial network to learn a mapping from input to output images. However, paired training examples are not sufficient. Zhu J Y et al. proposed CycleGAN [3], which is applied to unpaired image-to-image translation. CycleGAN uses a cycle consistency loss to preserve key attributes between the input and the translated image. However, these models usually map between two domains, and they are not practical to a large number of domains. Zhu J Y et al. proposed BicycleGAN [4] to handle mode collapse and produce more diverse results. Huang et al. proposed a multimodal unsupervised image-to-image translation (MUNIT) framework [5]. MUNIT produced high-quality and diverse results compared to those of BicycleGAN.

To address the scalability, several studies have proposed a unified framework. Choi et al. proposed StarGAN [4], which can learn the mapping relations among multiple domains using only a single model because StarGAN only generates the same output given an input. Choi et al. proposed StarGAN v2 [6], a scalable approach that can generate diverse images across multiple domains.

2.2. Neural Style Transfer

Gatys et al. [7] proposed Neural Style Transfer to synthesize an image with content similar to a given image and style similar to another. They define style feature by gram matrix. Li and Wand [8] combined Markov random fields and deep convolutional neural networks for synthesizing images with increased visual plausibility. However, these methods are too slow. To speed up the stylization, Johnson et al. [9] proposed perceptive loss function to train neural networks. Ulyanov et al. [10] proposed ways to improve the quality and diversity of the generated samples. However, the above methods aim to transfer fixed styles to content images. To address the fixed style problem, Li et al. [11] proposed a feed-forward architecture that can synthesize up to 300 textures and transfer 16 styles. Still, it couldn't adapt to arbitrary styles that are not observed during training. Huang and Belongie presented a novel adaptive instance normalization layer to enable arbitrary style transfer in real-time for the first time [12]. Soon after, Li et al. [13] presented a simple, yet effective method that is applied in universal style transfer.

2.3. Font Effect Transfer

Font effect transferring is related to the style transfer task. In 2017, font effect transfer was first raised by Yang et al. [14]. Yang synthesized images by analyzing correlated positions on the glyph. This method is vulnerable to different characters. Azadi et al. presented multi-content GAN [15], which aims at English character effect transfer. However, multi-content GAN can only handle 26 capital letters. Soon after, Men et al. [16] proposed a common framework for interactive texture transfer, which preserves both local structures and visual richness. First, it is suitable for general style. Then, Men et al. [17] presented an example-based method of text dynamic effects transfer. Yang et al. proposed Texture Effect Transfer GAN [18] for stylization and destylization. Yang et al. proposed shape-matching GAN [1], which enables controllable stylization. Gao et al. proposed AGIS-Net [19] to transfer both shape and texture styles with only a few stylized samples.

3. Proposed Method

The task called by the font effect transfer is to decorate font with different patterns. After transferring, the font's shape and texture change together. However, the structure of some characters is too complicated and make the processed characters unrecognizable. This violates principles of font effect transfer. Therefore, our work focuses on keeping font recognizable while increasing font art. Our solution to this method is utilizing a multi resolution feature pattern. Based on multi resolution feature, our model could guarantee the recognizability of the font while transferring.

3.1. Data Preprocessing

Data preprocessing is the critical step in machine learning process. In addition to general data enhancement methods, we have designed data preprocessing methods suitable for the task of generating special effects of artistic fonts. Our model utilizes multi resolution style images borrowed from pyramid features [20]. As Figure 2 shows, style image is many times down-sampled to smaller resolution images. These images are stacked together to form multi resolution style images. By using multi resolution style images, the details of the transferred characters are boosted. Different style features are helpful to generate different scale style shapes and textures. Its purpose is mainly to strengthen the ability of the model to process complicated glyphs.

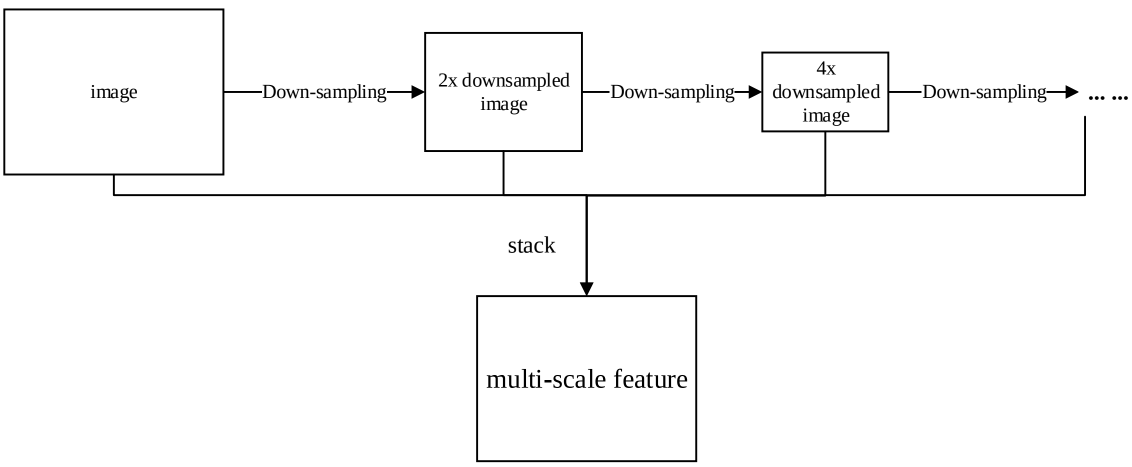


Figure 2. Process of multi-scale feature generation

One of the advantages of pyramids feature is that the generation of style effect could use a course-to-fine method. It is able to avoid produce course texture in fine places. For example, when the structure of a character is very complicated, if you use a high-resolution style image, the model does not know how to generate details. Certainly, low resolution features are not enough. In the end, it is best to design a pyramid input. It's a simple, yet effective method.

Besides multi resolution style images, our model utilizes angle jitter to expand style feature diversity. It obtains different direction style patterns by rotating style images. The pattern is the basis of the bending style features for complicated Chinese characters. Furthermore, plenty of style features can reduce overfitting in deep neural networks.

Our model utilizes erosion and dilation when the text in the image is deformed. Erosion and dilation are morphological image processing operations. As shown in Figure 3, they could remove unnecessary adjoins while not destroying the glyph structure.

3.2. Network Architecture

As shown in Figure 4, our network consists of the structure transform network and texture transform network. Therefore, our transfer process consists of two stages. The first stage is where the structure transform network learns the map between input character images and character images with style shapes. Structure transform network transforms binary character images into binary character images with style shapes. The second stage is where the texture transform network learns the map between character images with style shapes and character images with style shapes and textures. The texture transform network transforms binary character images with style shapes into character images with style shapes and textures. Finally, binary character images are transformed into character images with the pattern of style images.

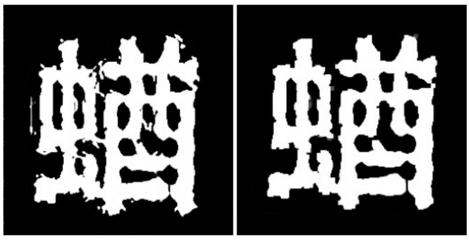


Figure 3. Remove unnecessary connection by utilizing erosion and dilation

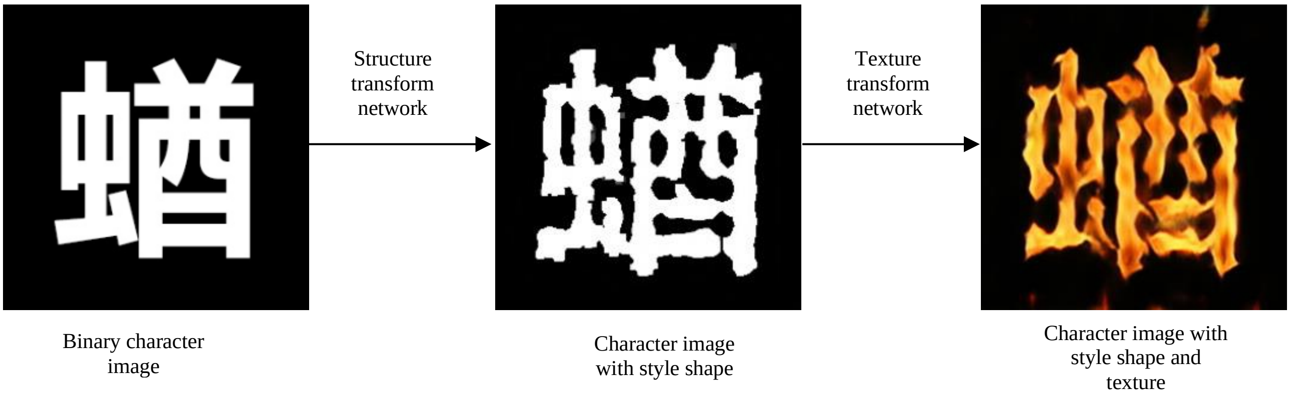


Figure 4. Overview of font effect generation process

As Figure 5 shows, the structure transform network consists of a generator network and discriminator network.

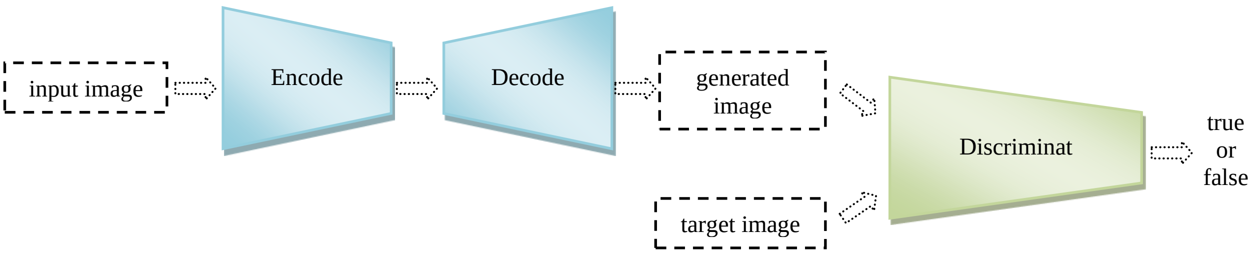


Figure 5. Structure transform network

In network structure, the texture transform network is similar to the structure transform network. The generator network is borrowed from CycleGAN [3]. The discriminator network is borrowed from PatchGAN [2].

3.3. Objective Function

(1) Reconstruction loss. This is the main component of our objective function. Its essence is L1 loss. It is used to calculate the distance between input and output at the pixel level. It could maintain structure and texture consistency between the output and target.

(2) Contextual loss. As illustrated in Equation (1), it focuses on the similarity of corresponding sets of features. It is not strictly calculating the pixel distance between two images. It effectively reduces the generation of artifacts. It is regarded as completion of reconstruction loss. Contextual loss is similar to style loss 9, however, it is evaluated that contextual loss is more suitable for our task.

$$contextualSimilarity(X,Y) = \frac{1}{N}\sum_{j}\max_{i}\ vggSimilarity_{ij} \tag{1}$$

X and *Y* are two images used to calculate the similarity. *N* is the count of VGG features. *ContextualSimilarity* represents contextual loss borrowed from 23. *VggSimilarity* represents feature produced by the VGG neural network.

(3) Adversarial loss. This is a key component of the GAN neural network, which can make the output more realistic and higher quality.

4. Experiment

4.1. Dataset

Data is from TE141K [21]. TE141K is extensively utilized in text effect transfer. Text effects are combinations of visual elements such as outlines, colors and textures of text. TE141K is split into three classes, including the English alphabet, Chinese characters and special symbols. Each class has different styles. There are 141,081 glyphs and 152 styles in total.

4.2. Comparison with Baseline

We use Shape-Matching GAN [1] as the baseline and conduct a number of experiments. As shown in Figure 6 and Figure 7, our method has superiority against the baseline at stylizing complex glyphs. On one hand, our method could produce appealing artistic effects. On the other hand, stylized character images can still be recognized.

Fire style image | Binary character image | Baseline results | Our results

Figure 6. Experiment results based on fire style image

4.3. Failure Cases

Our model doesn't always produce pleasant results. Since some style images are very complicated and are from the real world, our model may not effectively obtain information from style images or may obtain wrong information.

For example, when our model is applied to the Sakura style image shown in Figure 8, there exist some artifacts around the glyph shown in Figure 9. This will affect the feeling of appreciation of artistic fonts, although artifacts have no effect on character recognizability.

Figure 7. Experiment results based on maple style image

Figure 8. Sakura style image

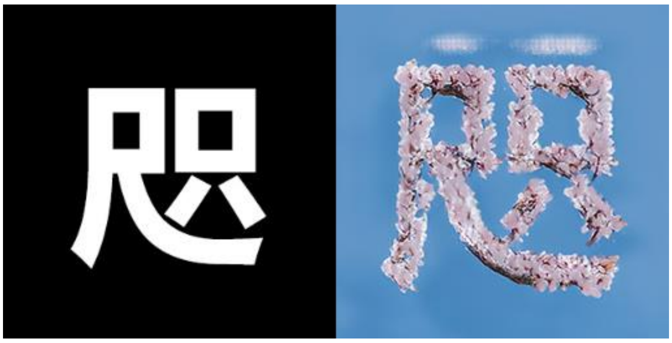


Figure 9. Failure case

4.4. Applications

Our model can be applied in the field of art design and typography. Our model can greatly increase efficiency. Besides, our model can provide more inspiration, which supports art designers to create more appealing works. Furthermore, it can also provide ideas for other style transfer tasks.

5. Conclusion and Future Work

We propose a method to improve font effect based on a multi resolution style feature. This effectively resolves the unrecognizability of complex glyphs with effects. Besides, we utilize a morphology operation to improve font effect further. Last but not least, we utilize contextual loss to calculate the distance between two images that are not strictly aligned. These methods improve the font effect generation model to produce pleasant results.

However, there are still limitations. In the future, we plan to utilize the attention mechanism to reduce training costs. It could make the model learn more effectively. Furthermore, we could focus on how to avoid the generation of artifacts when style patterns are really complicated.

Acknowledgements

This work was supported by the National Key Research and Development Project of China under Grant No. E0M2040101. We would also like to thank all the colleagues and graduate students who helped us with our system and experiments.

References

1. S. Yang, Z. Wang, Z. Wang, N. Xu, J. Liu, and Z. Guo, "Controllable Artistic Text Style Transfer via Shape-Matching GAN," in *Proceedings of 2019 IEEE/CVF International Conference on Computer Vision* (*ICCV*), pp. 4441-4450, Seoul, Korea (South), 2019
2. P. Isola, J. Zhu, T. Zhou, and A. A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks," in *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), pp. 5967-5976, Honolulu, HI, 2017
3. J. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks," in *Proceedings of 2017 IEEE International Conference on Computer Vision* (*ICCV*), pp. 2242-2251, Venice, 2017
4. J. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, et al., "Toward Multimodal Image-to-Image Translation," in *Proceedings of Advances in Neural Information Processing Systems Conference*, pp. 465-476, 2017
5. X. Huang, M. Liu, S. Belongie, and J. Kautz, "Multimodal Unsupervised Image-to-Image Translation," in *Proceedings of the European Conference on Computer Vision* (*ECCV*), pp. 172-189, 2018
6. Y. Choi, Y. Uh, J. Yoo, and J. W. Ha, "Stargan v2: Diverse Image Synthesis for Multiple Domains," in *Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8188-8197, 2020
7. L. A. Gatys, A. S. Ecker, and M. Bethge, "Image Style Transfer using Convolutional Neural Networks," in *Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), pp. 2414-2423, Las Vegas, NV, 2016
8. C. Li and M. Wand, "Combining Markov Random Fields and Convolutional Neural Networks for Image Synthesis," in *Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), pp. 2479-2486, Las Vegas, NV, 2016
9. J. Johnson, A. Alahi, and F. Li, "Perceptual Losses for Real-Time Style Transfer and Super-Resolution," in *Proceedings of the European Conference on Computer Vision* (*ECCV*), pp. 694-711, 2016
10. D. Ulyanov, V. Lebedev, A. Vedaldi, and V. S. Lempitsky, "Texture Networks: Feed-forward Synthesis of Textures and Stylized Images," *ICML*, Vol. 1, No. 2, p. 4, 2016
11. Y. Li, C. Fang, J. Yang, Z. Wang, X. Lu, and M. Yang, "Diversified Texture Synthesis with Feed-Forward Networks," in *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), pp. 266-274, Honolulu, HI, 2017
12. X. Huang and S. Belongie, "Arbitrary Style Transfer in Real-Time with Adaptive Instance Normalization," in *Proceedings of 2017 IEEE International Conference on Computer Vision* (*ICCV*), pp. 1510-1519, Venice, 2017
13. Y. Li, C. Fang, J. Yang, Z. Wang, X. Lu, and M. H. Yang, "Universal Style Transfer via Feature Transforms," in *Proceedings of Advances in Neural Information Processing Systems Conference*, pp. 386-396, 2017
14. S. Yang, J. Liu, Z. Lian, and Z. Guo, "Awesome Typography: Statistics-based Text Effects Transfer," in *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), pp. 2886-2895, Honolulu, HI, 2017
15. S. Azadi, M. Fisher, V. Kim, Z. Wang, E. Shechtman, and T. Darrell, "Multi-Content GAN for Few-Shot Font Style Transfer," in *Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7564-7573, Salt Lake City, UT, 2018
16. Y. Men, Z. Lian, Y. Tang, and J. Xiao, "A Common Framework for Interactive Texture Transfer," in *Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6353-6362, Salt Lake City, UT, 2018
17. Y. Men, Z. Lian, Y. Tang, and J. Xiao, "DynTypo: Example-based Dynamic Text Effects Transfer," in *Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), pp. 5863-5872, Long Beach, CA, USA, 2019

18. S. Yang, J. Liu, W. Wang, and Z. Guo, "TET-GAN: Text Effects Transfer via Stylization and Destylization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33, No. 1, pp. 1238-1245, July 2019
19. Y. Gao, Y. Guo, Z. Lian, Y. Tang, and J. Xiao, "Artistic Glyph Image Synthesis via One-Stage Few-Shot Learning," *ACM Transactions on Graphics*, Vol. 38, No. 6, pp. 1-12, November 2019
20. K. He, X. Zhang, S. Ren, and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 37, No. 9, pp. 1904-1916, September 2015
21. S. Yang, W. Wang, and J. Liu, "TE141K: Artistic Text Benchmark for Text Effect Transfer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020

Feiyu Zhang is a graduate student at the University of Chinese Academy of Sciences. His research interests include style transfer and deep learning.

Yi Yang is an Assistant professor at the Institute of Automation, Chinese Academy of Sciences. His research interests include data mining and visualization.

Weixing Huang is an Assistant professor at the Institute of Automation, Chinese Academy of Sciences. Her research interests include literature digitization.

Guigang Zhang is a Professor at the Institute of Automation, Chinese Academy of Sciences. His research interests include prognostics and health management.

Jian Wang is a Professor at the Institute of Automation, Chinese Academy of Sciences. His research interests include deep learning and control theory.