A Deep Image Compression Framework for Face Recognition

1st Nai Bian  
School of Microelectronics  
Xi’an Jiaotong University  
Xi’an, China  
biannai120@stu.xjtu.edu.cn

2nd Feng Liang  
School of Microelectronics  
Xi’an Jiaotong University  
Xi’an, China  
fengliang@xjtu.edu.cn

3rd Haisheng Fu  
School of Microelectronics  
Xi’an Jiaotong University  
Xi’an, China  
fhs4118005070@stu.xjtu.edu.cn

4th Bo Lei  
School of Microelectronics  
Xi’an Jiaotong University  
Xi’an, China  
lei_bo@stu.xjtu.edu.cn

Abstract—Face recognition technology has advanced rapidly and has been widely used in various applications. Due to the huge amount of data of face images in large-scale face recognition tasks and the large computing resource cost required correspondingly, there is a requirement for a face image compression approach that is highly suitable for face recognition tasks. In this paper, we propose a deep convolutional autoencoder compression network for face recognition tasks. In compression process, deep features are extracted from the original image by a Compression Network (CompNet) to produce a compact representation of the original image, which is then encoded and saved by an existing codec PNG. In reconstruction process, this compact representation is utilized by a Reconstruction Network (RecNet) to generate a restored image of the original one. In order to improve the face recognition accuracy when the compression framework is used in a face recognition system, we combine the CompNet and RecNet with an existing face recognition network for joint optimization. We test the proposed scheme and find that after joint optimization, the Labeled Faces in the Wild (LFW) dataset compressed by our compression framework has higher face verification accuracy than that compressed by JPEG2000, and is much higher than that compressed by JPEG.

Index Terms—face images compression, face recognition, convolutional autoencoder

I. INTRODUCTION

As a prominent representative of the well developed deep learning technology, face recognition has been successfully used in practical applications such as face identification on mobile phones and urban smart surveillance system. Large-scale face recognition tasks in such surveillance system analyze the images captured by cameras in various places to identify faces. In this process, the system needs to compare the obtained image with hundreds of millions of face images in its database. Such a large number of face images not only occupy a large amount of storage resource, but also expend a large amount of computing resource. So it is of great significance to propose an efficient image compression approach that can be adapted to tasks like face recognition and image classification.

As illustrated in Fig. 1, the process of a face recognition system includes two parts: preservation and processing of face images. The original image data are compressed, encoded, decoded and restored by the image codec. Then the restored image is processed by the face recognition network to obtain a series of deep features. The deep features here are often in the form of first order tensors. The basic work of face recognition is to compare the difference between the deep features of two face images.

Face images in many face recognition applications are compressed and reconstructed by conventional schemes such as JPEG. But there would be some annoying blocking artifacts in a picture when compressed at low bit rates by JPEG, which is presented in Fig. 2. Such artifacts not only affect image quality, but also affect the performance of feature extraction in subsequent image processing tasks such as face recognition and image classification.

Fig. 1. The components of face recognition system.

Fig. 2. Blocking artifacts of images compressed by JPEG at low bit rates.
Some post-processing methods such as Artifacts Reduction Convolutional Neural Network [2] (AR-CNN) are indeed useful for deblocking and artifacts reduction, which is shown in Fig. 3. However, these deblocking networks usually involve some iterative process that consumes high computational resource. Moreover, the optimization goal of the deblocking network is not entirely consistent with that of the face recognition network, that is to say, training a network that enhances image quality alone does not necessarily mean achieving the optimal result of face recognition tasks.

Recent years, there have been many learning-based image compression approaches showing excellent performance and some methods [3]–[6] even perform better than JPEG, JPEG2000 and BPG (a state of the art lossy coding method). Besides, due to its neural network structure and characteristic of backpropagation, a learning-based approach is possible to be combined with another face recognition network and trained jointly. Inspired by [1], in this paper, we replace the conventional codec in Fig. 1 with a compression-reconstruction network in the form of autoencoder combined with a standard lossless image codec PNG. They are combined to realize the functional requirement of image compression and recovery. We directly connect this compression-reconstruction framework with a face recognition network.

In this system, a face image is input into CompNet to produce a compact representation of the original image. The compact representation is 1/8 of the size of the original image and is encoded and decoded with PNG (a standard lossless image codec). This compact representation is then sent to RecNet to restore the original image. The reconstructed image passes through a face recognition network to generate a series of feature vectors, which is called the deep features of the face. These feature vectors can be used to implement face recognition through a multiclassifier such as LMCL [11] (Large Margin Cosine Loss, the classifier and loss function used in this paper), or directly compare with that of another face image. We jointly train the CompNet and RecNet, so that the compact representation produced by CompNet can preserved the useful information of the original image as much as possible and that the reconstructed image generated by RecNet can be restored as close as possible to the original image. We then connect the proposed compression-reconstruction network with another existing face recognition network, set the pre-trained model parameters of both networks as initialization and jointly train it. We assume that the parameters of the combined network will converge faster in this way than being fully retrained in view of the idea of transfer learning.

II. CNN BASED FACE IMAGE COMPRESSION-RECONSTRUCTION-RECOGNITION JOINT FRAMEWORK

A. The overall structure of the joint framework

The overall structure of the proposed compression-reconstruction-recognition joint framework is illustrated in Fig. 4. We replace the conventional image compression codec in the face recognition system with a deep compression network composed of two convolutional neural networks called Compression Network (CompNet) and Reconstruction Network (RecNet) and a standard lossless image codec PNG. They are combined to realize the functional requirement of image compression and recovery. We directly connect this compression-reconstruction framework with a face recognition network.

In this system, a face image is input into CompNet to produce a compact representation of the original image. The compact representation is 1/8 of the size of the original image and is encoded and decoded with PNG (a standard lossless image codec). This compact representation is then sent to RecNet to restore the original image. The reconstructed image passes through a face recognition network to generate a series of feature vectors, which is called the deep features of the face. These feature vectors can be used to implement face recognition through a multi-classifier such as LMCL [11] (Large Margin Cosine Loss, the classifier and loss function used in this paper), or directly compare with that of another face image. We jointly train the CompNet and RecNet, so that the compact representation produced by CompNet can preserved the useful information of the original image as much as possible and that the reconstructed image generated by RecNet can be restored as close as possible to the original image. We then connect the proposed compression-reconstruction network with another existing face recognition network, set the pre-trained model parameters of both networks as initialization and jointly train it. We assume that the parameters of the combined network will converge faster in this way than being fully retrained in view of the idea of transfer learning.

B. Design of CompNet and RecNet

1) Network structure of CompNet and RecNet: The architectures of CompNet and RecNet is illustrated in Fig. 5. It is designed in the form of convolutional autoencoder. CompNet and RecNet are symmetrical designed. As shown in Fig. 5,
CompNet uses a five-layer convolution structure. The second, third, and fourth convolutional layers are called downsampling layers, whose strides are set to 2. Each time a downsampling layer is passed, the height and width of the feature maps become 1/2 of the original ones. There are 3 downsampling layers, so the height and width of the final compact map are both 1/8 of the original image.

Correspondingly, there are 3 upsampling layers using de-convolution structure on the RecNet side. Through every upsampling layer, the height and width of feature maps will double, so that a original-sized image can be restored from a compact map of 1/8 of its height and width. We add a residual block between every two upsampling layers and every two downsampling layers. In addition, inspired by [8], we change the structure of the residual block and add the dropout module to get better results. The structure of the residual block is shown in Fig. 6. In order to achieve faster convergence speed, we replace the default Rectified Linear Unit(ReLU) with a Parametric Rectified Linear Unit(PReLU).

2) Quantization and entropy coding: The values of the original image are integers in [0, 255]. In order to train the network better, all image data will be normalized to floating-point number in [-1, 1] before input into the network. The values of the compact map are still floating-point numbers in [-1, 1]. In order to encode the compact representation whose values range in [-1, 1] with existing codecs, it is necessary to first map those values to [0, 255] and quantify them:

\[ X_Q = \text{round} \left( \frac{(X_E + 1) \times 255}{2} \right) \quad (1) \]

where \( X_E \in [1, 1] \) represents the compact map produced by CompNet, \( X_Q \in [0, 255] \) represents the image after quantization.

In practical applications, we use a common standard lossless image codec PNG to encode the compact map. Before input into RecNet, the values of the compact map need to be converted to floating-point number in [-1, 1] again. That is, \( (X_Q \times 2/255 - 1) \) is input into RecNet to obtain a reconstructed image. Quantization is a zero-gradient operation so the gradient of model parameters would be zero, thus these model parameters would not be able to be updated. As a result, the quantization part cannot be directly put into the network during training. But let’s assume that the quantization operation only introduces a negligible error, so we skip the quantization function in training.

3) Training of CompNet and RecNet: We use the open source dataset published by the Computer Vision Lab of ETH Zurich as the training set for our CompNet and RecNet. The reason why we didn’t use a face image dataset as training set is that there are few high-definition face datasets, but there is a high requirement on the image definition to train an image compression network. To better train our network, we use several data enhancement methods, such as rotation and scaling. We extract 81,650 image blocks from 1633 high-quality images. These image blocks are losslessly saved as PNG format to avoid compression artifacts. We only use mean square error (MSE) as the loss function:

\[ L = \frac{1}{N} \|Y_n - X_n\|^2 \quad (2) \]

where \( N \) represents batchsize, \( X_n \) represents the original image, \( Y_n \) represents the restored image which is the output of RecNet.

CompNet and RecNet are trained together using the Stochastic Gradient Descent (SGD) as the optimizer. We set batchsize to 20 and train the network for 40 epochs with a fixed learning rate of 0.0001.

C. Selection and pre-training of face recognition network

In recent years, convolutional neural networks (CNN) based face recognition approaches have well developed, and some [9]–[12] have achieved more than 99% verification accuracy on the LFW dataset utilizing different loss functions respectively. Given the high efficiency of the existing face recognition works, we directly utilize one of these sophisticated solutions and add it to our compression-reconstruction framework. In our work, we use the scheme proposed in Cosface [11]. It uses sphere20 as its network structure, and adopts LMCL (Large Margin Cosine Loss) as its loss function. It was pre-trained on CASIA-WebFace dataset, and can achieve an verification accuracy of 99.28% on the LFW dataset.

1) Structure of Cosface Face Recognition Model: Cosface’s sphere20 network are shown in Fig. 7. The pre-trained model parameters of the feature extraction part which is outside the
dotted box in Fig. 7 have been given. The parameters of the classifier part which is inside the dotted box in Fig. 7 have to be retrained. The pre-trained model parameters of the face feature extraction part are imported and frozen during training process. This part does not participate in backpropagation. Only the parameters of the classifier are updated in backpropagation. 452,960 face images from 10,575 individuals in CASIA-WebFace dataset are cropped by Multi-task Cascaded Convolutional Networks (MTCNN, a face detection and calibration framework) [13] to the size of 112 × 96 and then are used as the training set. Batchsize is set to 100. The gradient descent algorithm is SGD. Learning rate is initially set to 0.1 and gradually reduced to 0.0000001. After 6 epochs of training, we obtained a pre-trained model of the classifier of the Cosface network.

D. Joint training of the combined network

During training process, our combined network omits the quantization and entropy coding for compact representation, due to the same reason mentioned already. As shown in Fig. 8, we directly input the compact representation of the output of CompNet to RecNet. Before the original image input into CompNet, we first normalize its pixel values from [0, 255] to [-1, 1]. The loss function of the combined network is still the loss function of the Cosface, that is LMCL. This is because we use the accuracy of face recognition as the only optimization goal and ignore the image restoration effect in joint training.

Since we have the pre-trained models and parameters of CompNet, RecNet, feature extraction part and classifier part of Cosface, they are all imported into the corresponding locations in the combined network. We use the same CASIA-WebFace dataset of the size of 112 × 96 as training set. Batchsize is set to 100, gradient descent algorithm is SGD, learning rate is 0.01 initially and then gradually reduced to 0.00001. The combined network is trained for 40 epochs.

III. EXPERIMENT

A. Image recovery effect of Initially trained CompNet and RecNet

After completing the pre-training of CompNet and RecNet, we first use the dataset released by Kodak PhotoCD to roughly test its performance of image reconstruction. Fig. 9 shows the image recovery effect of JPEG, JPEG2000, and our initially trained CompNet and RecNet at the same bits per pixel (bpp). Subjectively speaking, the images compressed by JPEG have serious blocking artifacts. Images compressed by the proposed CompNet and RecNet that we have initially trained are somewhat ambiguous. JPEG2000 compressed images have the best recovery effect among three methods. However, the recovery effect of different compression schemes at the same bit rates is not the main goal pursued in this paper. Therefore, we did not intend to compared the restoration effect of the images compressed with these three compression schemes in detail by objective indicators (such as PSNR, SSIM).

B. Comparison of face verification accuracy of LFW datasets compressed by different compression methods

Our only focus in this paper is on the recognition accuracy when applying these compression schemes to the face recognition system. To accomplish this purpose, we fist use MTCNN to process face detection and calibration on the LFW dataset, and generate 13233 face images of 5749 people with the picture size of 112×96 as the benchmark test set. For this LFW_112×96 dataset, we respectively compress it by JPEG, JPEG2000 and the proposed CompNet that has been jointly trained with the face recognition network to the same size, and then test the face verification accuracy on these datasets. (By the way, even with the smallest quality factor 0 when compressed by JPEG, the LFW_112×96 dataset can still not be compressed smaller than 10.8MB.) In addition, in order to evaluate the effect of joint optimization, we also test the verification accuracy of the LFW_112×96 dataset compressed by our proposed CompNet before jointly trained and by the JPEG2000 scheme with the same compression ratio respectively. The experimental results are shown in Table I.
Looking horizontally, the original LFW\textsubscript{112×96} dataset can achieve a face verification accuracy of 99.28%. The size of the dataset compressed by CompNet after jointly trained becomes 1/7 of the original one, and the face verification accuracy only decreases by 9.8%, reaching 89.48%, which is 3.55% higher than the accuracy of 85.93% of the JPEG2000 compressed dataset with the same compression ratio, and far more higher than the poor accuracy of 67.87% of the JPEG compressed dataset. In addition, when our pre-trained network (before jointly trained) compresses the LFW\textsubscript{112×96} dataset to 2.79MB (only 1/11 of the original dataset), the face verification accuracy is higher than the JPEG2000 method under the same compression ratio, and still higher than the JPEG method.

Looking vertically, joint training has significantly improved the face verification accuracy (from 75.87% to 89.48%), indicating that joint optimization aiming at improving face recognition accuracy is indeed effective. For analysis, we present some of the face images in the LFW\textsubscript{112×96} dataset recovered by different compression schemes in Fig. 10.

As can be seen from Fig. 10, under the same compression ratio, the face images restored by the proposed network after jointly trained (4.38MB) are reddish and its restoration effect is worse than that compressed by JPEG2000 but get a higher accuracy. We think the reason why the pictures are reddish is that it’s grey information instead of color information that matters in face recognition so these images mainly retain information of channel R and abandon that of channel G and B. As for the reason why our method has higher accuracy, we hold the view that on the one hand, through joint training, the spatial structure of the values of reconstructed images may be adapted to the goal of face recognition, on the other hand, we have observed that in images of JPEG2000 compressed dataset (4.41MB) there are obvious discontinues color patches in the restored images. The abrupt boundary between these color patches is likely to be recognized by the feature extraction network as an “edge” feature, which will interfere the extraction of faces’ truly essential features. Images compressed by the proposed jointly trained network (4.38MB) however does not have the above-mentioned color patches. Although they are reddish, they highlight the important parts of faces. Despite the images restored by the proposed network after jointly trained differ a lot in color from original images and nearly can not be recognized by human, in practical surveillance system which is highly automatic, pictures are processed by computers rather than human eyes, so the recovery effect of images is actually secondary.

In addition, the face images recovered by the proposed network before jointly trained (2.79MB) is blurry, but still
have higher accuracy than that recovered by JPEG (10.8MB) which have serious blocking artifacts and that recovered by JPEG2000 (2.78MB) which has much more serious discontinuous color patches. The reason can still be explained as that false boundaries caused by discontinuous blocking artifacts and color patches are affecting the extraction of the truly essential face features.

In order to know the influence of the error caused by the quantization of the compact map on face recognition, we compare the verification accuracy of LFW dataset compressed by the proposed network before and after jointly trained with and without quantization. The results are shown in Table II. It can be seen that without quantization, that is, the compact map that produced by CompNet is directly input into the RecNet and the face recognition network, the accuracy of the face verification is higher than that with quantization by 2%–3%. This is because there is error between the rounding results after quantization process and the original compact map’s values. If we make some changes to the quantization process, there’s still space for improvement in the face recognition performance.

### IV. CONCLUSION

In this paper, we propose a learning based compression and reconstruction framework in the form of convolutional autoencoder, and combine it with a standard lossless image codec PNG. We connect them with a existing face recognition network and train them jointly. Through experiment, we find that under the same compression ratio, the face verification accuracy of the LFW dataset compressed by our compression scheme is higher than that of the JPEG2000 scheme, and much higher than that of the JPEG scheme. In the future, we will continue to explore some methods to simulate the quantization process to further improve the performance of face recognition.

#### ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China under Grant No. 61474093 and by Tencent.

### TABLE II

| Compression Network | Verification Accuracy on LFW |
|---------------------|-------------------------------|
|                      | With quantization | Without quantization |
| Our CompNet before jointly trained | 78.37% | 75.87% |
| Our CompNet after jointly trained | 92.23% | 89.48% |

[1] F. Jiang, W. Tao, S. Liu, J. Ren, X. Guo and D. Zhao, “An End-to-End Compression Framework Based on Convolutional Neural Networks,” in IEEE Transactions on Circuits and Systems for Video Technology, vol. 28, no. 10, pp. 3007-3018, Oct. 2018.

[2] C. Dong, Y. Deng, C. C. Loy and X. Tang, “Compression Artifacts Reduction by a Deep Convolutional Network,” 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, 2015, pp. 576-584.

[3] E. Agustsson, F. Mentzer, M. Tschannen, L. Cavigelli, R. Timofte, L. Benini, and L. V. Gool, “Soft-to-hard vector quantization for end-to-end learning compressible representations,” in Proc. Adv. Neural Inf.Process. Syst. 2017, pp. 1141-1151.

[4] E. Agustsson, M. Tschannen, F. Mentzer, R. Timofte, and L. V. Gool, “Generative adversarial networks for extreme learned image compression,” arXiv preprint arXiv:1804.02958, Apr. 2018.

[5] L. Theis, W. Shi, A. Cunningham, and F. Huszar, “Lossy image compression with compressive autoencoders,” in Int. Conf. Learning Representations, 2017.

[6] G. Toderici, D. Vincent, N. Johnston, S. J. Hwang, D. Minnen, J. Shor, and M. Covell, “Full resolution image compression with recurrent neural networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recog., 2017, pp. 54355443.

[7] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recog., June 2016, pp. 770778.

[8] H. Liu, T. Chen, Q. Shen, T. Yue, Z. Ma, “Deep Image Compression via End-to-End Learning,” arXiv preprint arXiv:1806.01496, 2018.

[9] F. Wang, X. Xiang, J. Cheng, and A. L. Yuille, “Normface: L2 hypersphere embedding for face verification,” in Proceedings of the 2017 ACM on Multimedia Conference, MM 2017, Mountain View, CA, USA, October 23-27, 2017, pp. 1041-1049, 2017.

[10] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj and L. Song, “SphereFace: Deep Hypersphere Embedding for Face Recognition,” in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 6738-6746.

[11] H. Wang et al., “CosFace: Large Margin Cosine Loss for Deep Face Recognition,” in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, 2018, pp. 5266-5274.

[12] J. Deng, J. Guo, and S. Zaferiou, “Arcface: Additive angular margin loss for deep face recognition,” arXiv:1801.07698, 2018.

[13] K. Zhang, Z. Zhang, Z. Li and Y. Qiao, “Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks,” in IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499-1503, Oct. 2016.