QNet: An Adaptive Quantization Table Generator Based on Convolutional Neural Network

Xiao Yan, Yibo Fan, *Member, IEEE*, Kewei Chen, Xulin Yu, and Xiaoyang Zeng, *Member, IEEE*

***Abstract*— The JPEG is one of the most widely used lossy image-compression standards, whose compression performance depends largely on a quantization table. In this work, we utilize a Convolutional Neural Network (CNN) to generate an image-adaptive quantization table in a standard-compliant way. We first build an image set containing more than 10,000 images and generate their optimal quantization tables through a classical genetic algorithm, and then propose a method that can efficiently extract and fuse the frequency and spatial domain information of each image to train a regression network to directly generate adaptive quantization tables. In addition, we extract several representative quantization tables from the dataset and train a classification network to indicate the optimal one for each image, which further improves compression performance and computational efficiency. Tests on diverse images show that the proposed method clearly outperforms the state-of-the-art method. Compared with the standard table at the compression rate of 1.0 bpp, the regression and classification network provide average Peak Signal-to-Noise Ratio (PSNR) gains of nearly 1.2 and 1.4 dB. For the experiment under Structural Similarity Index Measurement (SSIM), the improvements are 0.4% and 0.54%, respectively. The proposed method also has competitive computational efficiency, as the regression and classification network only take 15 and 6.25 milliseconds, respectively, to process a 768×512 image on a single CPU core at 3.20 GHz.**

***Index Terms*— Convolutional neural network (CNN), image compression, JPEG, quantization table, peak signal-to-noise ratio (PSNR), structural similarity index measurement (SSIM).**

I. Introduction

SINCE its proposal by the Joint Photographic Experts Group in 1992 [1], the JPEG has become the most commonly used lossy image compression standard. Compared to other image compression formats, the JPEG has a great advantage in flexibility, as it can easily adjust the degree of compression. Although some later lossy image-compression standards, such as JPEG 2000, WebP, and BPG, have in some ways exceeded its performance, the JPEG is still the most popular image format used in the World Wide Web and digital cameras due to its flexibility and universality.

A conventional JPEG encoder has three basic steps. For each color component, the image is divided into 8×8 blocks and transformed to 8×8 coefficient matrices via 2D Discrete Cosine Transform (DCT). The resulting DCT coefficients are uniformly quantized by an 8×8 quantization table and transferred to an entropy encoder. The JPEG standard allows the encoder free selection of quantization tables and Huffman tables. Thus the optimization research for JPEG compression mainly focuses on Huffman table optimization [2]–[6], quantization table optimization [7]–[16], and DCT coefficient adjustment [17], [18], aiming to achieve better image quality at the same or a higher compression rate.

Many studies have shown that the quantization table is the key factor in compression performance. Research focuses on many methods of quantization table optimization, such as rate-distortion optimization [7]–[11], human visual system (HVS)-based optimization [12]–[14], heuristic optimization [15], [16], and Deep Neural Network (DNN) favorable optimization [25]. Optimization methods can generally be divided into two categories. One is to propose a new universal quantization table [12], [16], [25], and the other to adaptively generate the optimal quantization table according to the input image [7]–[11], [13]–[15]. A universal quantization table has difficulty performing well on all kinds of images due to their diverse content, while the generation of a self-adaptive quantization table often results in complex iterative operations.

To obtain a self-adaptive quantization table for each image without iterative calculation, we propose a CNN-based optimal quantization table generator, QNet, to learn the correspondence between an image's features and its optimal quantization table, and then infer the optimal quantization table efficiently for each input image according to its statistical characteristic. Other optimization methods, such as soft threshold quantization and Huffman table optimization, can be used in conjunction with this method.

Neural network based compressions have made a series of remarkable achievements, some even outperforming JPEG, WEBP, and BPG [19]–[21], [26]. Johnston *et al.* [19] presented a general architecture for compressing with RNNs, content-based residual scaling, and a new variation of GRU. Experimental results prove that the proposed architecture outperforms JPEG at image compression across most bitrates on the rate-distortion curve on the Kodak dataset images, with and without the aid of entropy coding. Huszar *et al.* [20] aimed at directly optimizing the rate-distortion tradeoff produced by

Manuscript received July 17, 2019; revised February 13, 2020, May 1, 2020, July 6, 2020, and September 3, 2020; accepted September 25, 2020. Date of publication October 16, 2020; date of current version October 22, 2020. This work was supported in part by Alibaba Innovative Research (AIR) Program, in part by the Shanghai Science and Technology Committee (STCSM) under Grant 19511104300, in part by National Natural Science Foundation of China under Grant 61674041, in part by the Innovation Program of Shanghai Municipal Education Commission, and in part by the Fudan University-CIOMP Joint Fund under Grant FC2019-001. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Yui-Lam Chan. *(Corresponding author: Yibo Fan.)*

Xiao Yan, Yibo Fan, Kewei Chen, and Xiaoyang Zeng are with the State Key Laboratory of ASIC and System, Fudan University, Shanghai 200433, China (e-mail: yanxiao@fudan.edu.cn; fanyibo@fudan.edu.cn; xyzeng@fudan.edu.cn; 17212020003@fudan.edu.cn).

Xulin Yu is with Alibaba Group, Hangzhou 311121, China (e-mail: xiangwu@alibaba-inc.com).

Digital Object Identifier 10.1109/TIP.2020.3030126

an autoencoder. A simple but effective approach was proposed for dealing with the non-differentiability of rounding-based quantization, and for approximating the non-differentiable cost of coding the generated coefficients. Their performance is similar to JPEG 2000 when evaluated for perceptual quality. Simoncelli *et al.* [26] presented a complete image compression method based on nonlinear transform coding, and a framework to optimize it end-to-end for rate-distortion performance. Using a variant of stochastic gradient descent, they jointly optimized the entire model for rate-distortion performance over a database of training images, introducing a continuous proxy for the discontinuous loss function arising from the quantizer. The compression method offers improvements in rate-distortion performance over JPEG and JPEG 2000 for most images and bit rates. Bourdev *et al.* [21] firstly used GANs for image compression and proposed a compression architecture that consist of an autoencoder featuring pyramidal analysis, an adaptive coding module, and regularization of the expected codelength. Their algorithm typically produces files 2.5 times smaller than JPEG and JPEG 2000, 2 times smaller than WebP, and 1.7 times smaller than BPG on datasets of generic images across all quality levels. At the same time, the codec is also lightweight and deployable. Most of the neural network-based compressions use similar architecture to autoencoders, which directly encode images in a variety of formats. Thus, they are relatively lacking in compatibility and are difficult to apply on a common platform. The proposed method simply utilizes the neural network to obtain an optimal quantization table, which is highly standard-compatible.

In this paper, we first build an image set containing more than 10,000 images and generate their optimal quantization tables through a computing-intensive genetic algorithm. After that, we propose a novel method to efficiently extract and fuse the frequency and spatial domain information of each image and train a regression network, namely QNet-R, to directly generate the quantization table. In training QNet-R, the quantization tables obtained by the genetic algorithm are used as the ground truth. We analyze the images and their optimal quantization tables and determine that images with similar frequency domain distributions often correspond to very similar optimal quantization tables. Inspired by this, we place the optimal tables into several categories by Principal Component Analysis (PCA) and a K-means clustering algorithm and mark the images with the category to which their optimal quantization table belongs. These cluster centers are used as representative quantization tables, and all we need is an efficient classification network, namely QNet-C, to indicate the right one for each input image. In training QNet-C, the category index of each image is used as the classification label.

QNet is independent of the quantization table optimization algorithm used to build the training set, so a variety of methods can be chosen according to the application scenario. We use the classical genetic algorithm to generate the optimal quantization table for images in the training set because of its relatively high accuracy. In this work, QNet is trained under PSNR and SSIM to prove that it is also independent of the image quality assessment. In practice, the image quality evaluation index is easily replaced by other HVS-based evaluation indices, such as Feature-Similarity Index Measurement (FSIM) [16].

The rest of this paper is organized as follows. Section II reviews previous quantization table optimization work in the literature. The CNN-based quantization table optimization method is described in section III. Experiments and analysis of results are presented in section IV. Conclusions are drawn in section V.

II. Related Work

A. Rate-Distortion Optimization

The rate-distortion optimization method establishes a rate-distortion function and then minimizes the total rate-distortion cost to obtain the optimal quantization table. The main drawback of traditional rate-distortion optimization is difficulty balancing computational cost and accuracy. Hence, the crucial problem in rate-distortion optimization is to more efficiently evaluate quantization tables [11]. Ratnakar and Livny [10] used the empirical entropy of quantized DCT coefficients to estimate the rate and employed histograms to calculate the rate and distortion to significantly improve JPEG compression performance. Yang *et al.* [11] modeled transform coefficients across different frequencies as independently distributed random sources and applied the Shannon lower bound to approximate the rate-distortion function of each source. They proposed an efficient statistical-model-based algorithm to design quantization tables for DCT-based image coding. Experimental results show that the work of Yang *et al.* [11] surpassed that of Ratnakar and Livny [10] in both PSNR improvement and computational performance and is currently the best rate-distortion optimization algorithm.

However, since they must establish a distortion function, current rate-distortion optimization methods are mainly based on Mean Square Error (MSE) and have difficulty achieving good performance under complex or subjective image quality assessment. In contrast, QNet merely learns the correspondence between the image and its optimal quantization table and is easily applied under various image quality assessments.

B. HVS-Based Optimization

HVS-based optimization attempts to model the human visual system and then generates an optimal quantization table according to the perceptual importance of each frequency component. Wang *et al.* [12] derived a perceptual quantization table to replace the standard table by incorporating the human visual system model with a uniform quantizer. Since this method attempts to provide another universal table, it incurs no additional computational cost, but it has limited generalization ability. Zhang *et al.* [14] used the Just-Noticeable Difference (JND) model to represent human-perceived distortion and obtained the optimal quantization table by minimizing the rate-distortion cost at all frequencies. By combining rate-distortion optimization with the HVS method, this method achieves significant rate improvement with high subjective image quality and a large computational cost, since it iteratively calculates the rate-distortion function.

Aiming mainly just to learn the correspondence between image features and optimal quantization tables, QNet is easily

combined with various HVS by establishing different training sets.

C. Heuristic Optimization

Heuristic optimization methods include simulated annealing algorithms, genetic algorithms, and particle swarm optimization. They usually start from a random solution and iterate to find the optimal solution constrained by a specific quality evaluation index. Manavalan *et al.* [15] proposed a knowledge-based genetic algorithm combining knowledge about image compression with traditional genetic algorithms, resulting in a better rate-distortion tradeoff and faster convergence speed compared to classical methods. However, it needs to evolve for hundreds of generations to obtain an approximate optimal solution, which means a huge computational cost.

Max *et al.* [16] used a simulated annealing algorithm to generate several quantization tables that outperform the standard table under FSIM. They claimed that these new quantization tables could reduce the distortion under FSIM by over 10% while improving compression by over 20% at quality level 95. However, the optimization tables in the paper are manually selected according to the compression result of a test set, which means that these tables may be difficult to adapt to various image contents. In addition, in compressing an image, the compressor does not know which of the optimal quantization tables to use.

D. DNN Favorable Optimization

Different from human perception-oriented optimization methods, Liu *et al.* [25] proposed a DNN-oriented quantization table design method, DeepN-JPEG, to decrease data offloading and local storage cost in terminal devices. Experimental results showed that DeepN-JPEG improved the compression rate by a factor of nearly 3.5 and consumed only 30% of the power of the conventional JPEG with no loss of classification accuracy. With the explosive growth of artificial intelligence applications, image compression tasks will increasingly be oriented toward neural networks rather than human perception.

In summary, current human perception-oriented JPEG quantization table optimization methods are limited by the shortcomings of the coupling with image quality assessment, lack of generalization, and high computational cost. Different from previous research, the QNet proposed in this work is independent of specific image quality assessments and can adaptively generate the optimal quantization table for each input image without significantly increasing the computational workload.

III. CNN Based Optimization

In this section, we first describe the network architecture to extract and fuse the features in the frequency and spatial domains, and we then introduce the regression network QNet-R and classification network QNet-C used in our research. The method of extracting representative quantization tables from the optimal quantization table space is also described.

A. Feature Extraction and Fusion

In the JPEG encoding process, the input image is partitioned into non-overlapping 8×8 blocks and transformed from the pixel domain to the frequency domain by a normalized, two-dimensional DCT transformation. After transformation, the image data matrix contains both the intra-block frequency domain information and inter-block spatial domain information.

The ordinary convolution layer slides the convolution kernels on the input matrix, which inevitably confuses the frequency domain information in different blocks and hinders the efficient extraction of meaningful feature maps. Thus, the critical problem is to make the information in these two domains fully usable while not confusing each other.

The convolution layer with kernel size 1×1 was first used to reduce the number of feature channels or cross-channel information interactions [22]. For the sake of brevity, hereafter we refer to the convolution layer with kernel size 1×1 as 1×1 convolution. In this work, we use 1×1 convolution to extract the intra-block frequency domain information without confusing it with the inter-block spatial domain information. As shown in Fig. 1, we consider the 64 frequency points in each block as 64 channels and use the 1×1 convolution layer to extract the intra-block feature in the frequency domain. This layer performs convolution on the same positions of the 64 channels, which only involves DCT coefficients in a single 8×8 block. After that, the ordinary convolution layer can be used to fuse each block feature in the spatial domain without information confusion. This method can be used by the image processing tasks with a fixed block size to avoid information confusion while extracting intra- and inter-block features.

B. QNet-R: Regression Network

A neural network can generally be built for any resolution. This paper takes 768×512 as an example, since it is representative and widely used. For other resolutions, we can resize images to this fixed resolution as needed. Fig. 2 shows the architecture of the regression network. We merely describe the luminance component of the image, since the quantization process of the three components of a JPEG image is the same, and a similar structure can be used for the chrominance components. The size of each feature map is marked between adjacent layers. The final output of QNet-R is a 1×64 vector, which represents an 8×8 quantization table.

As mentioned before, the input image is divided into 8×8 blocks, and then we apply a DCT transformation. We adopt a 1×1 convolutional layer to extract the frequency domain features from each block, which is followed by a rectified linear unit (ReLU) as an activation function. The local response normalization layer is connected after the 1×1 convolutional layer to improve the generalization capabilities of the entire network.

Since the size of the final output feature map is quite large (8×8), we first use a deep convolutional neural network to extract spatial features. However, deep networks are prone to overfitting, it is hard to pass gradient updates through the entire network [22], and to naively stack large convolutional layers is computationally expensive. An inception module [23] has an excellent topological structure, and it provides powerful representation ability without increasing the depth of the network. The architecture of the inception module with dimension

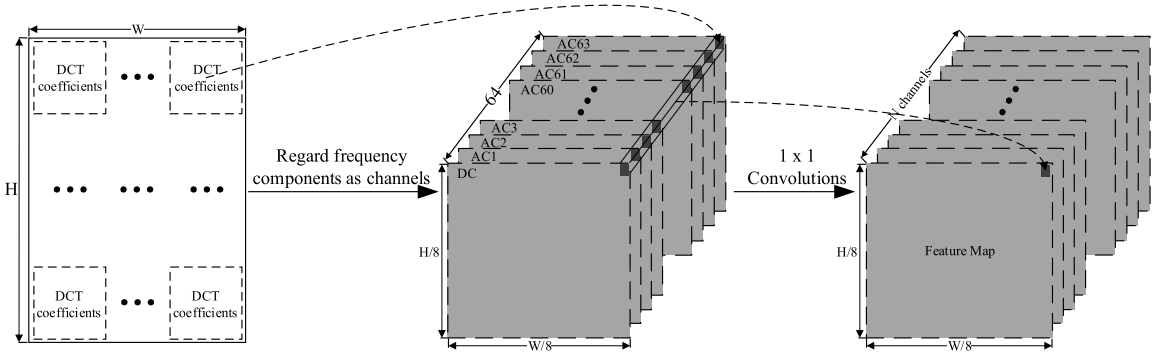


Fig. 1. A 2-dimensional DCT coefficient matrix of size (W, H) is converted to a tensor of size (W/8, H/8, 64) according to 64 frequency points, and then a 1×1 convolution operation is performed in the frequency dimension to extract the frequency domain characteristics.

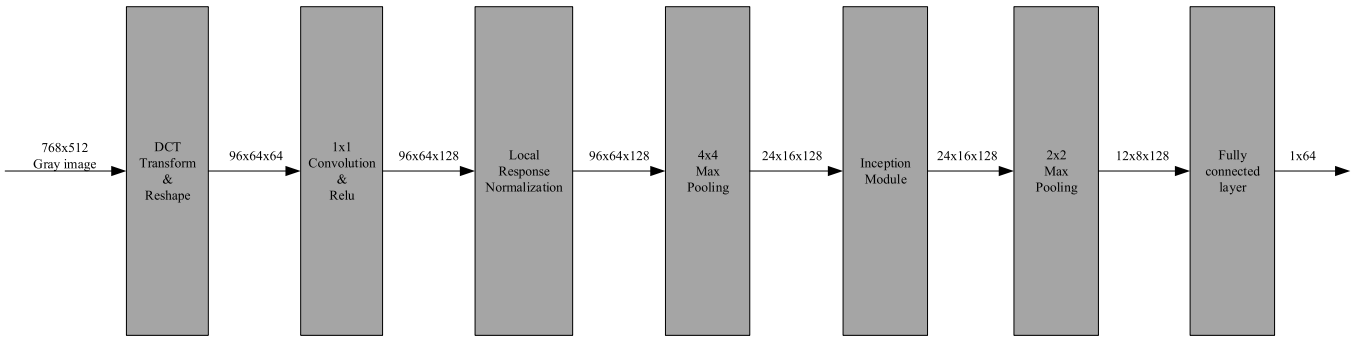


Fig. 2. The structure of the regression network.

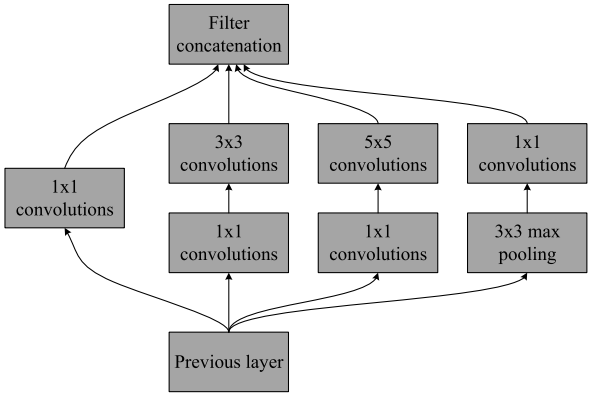


Fig. 3. The structure of inception module with dimension reductions. Compared to the naive version, 1×1 convolution is used to compute reductions before the expensive 3×3 and 5×5 convolutions as a function of applied field.

reductions is shown in Fig. 3. Unlike traditional CNNs that aim to stack the networks deeper, the inception module tries to widen the network. Here, it performs convolution on the input feature maps with three filter sizes (1×1, 3×3, 5×5). The outputs are concatenated and sent to the next layer. With the help of the inception module, we can fuse the spatial domain features with frequency domain features between all 8×8 blocks with little increase in computational complexity. The final output is obtained by a fully connected layer.

C. Representative Quantization Tables Extraction

During our research, we found that images with similar Alternating Component (AC) coefficient distributions often correspond to similar optimal quantization tables. Images with different frequency domain features are shown in Fig. 4, and the distributions of their AC coefficients and optimal quantization tables are shown in Fig. 5. By observing Fig. 5, we can find that the quantization tables corresponding to images in Fig. 4 (a)-(c), which have rich high-frequency components, are almost uniformly distributed in the frequency domain. But for those composed mainly of low-frequency components, as in Fig. 4 (d)-(f), the high-frequency entries are significantly larger than the low-frequency entries.

The above observation implies that perhaps just a few typical quantization tables can represent the entire optimal quantization table space. Hence, we might extract one or several representative quantization tables to simplify the adaptive quantization table generation network. Max *et al.* [16] obtained several optimal quantization tables from simulated annealing, but the tables were manually selected according to a small test set. We employ PCA and K-means clustering to place the optimal tables into several categories and label the images as the category to which their optimal quantization table belongs. The case that the quantization tables and images are placed into two categories is illustrated in Fig. 6. The cluster center quantization tables are used as the adaptive quantization tables for images in each category, so the entire optimal quantization table space can be simply represented by several cluster centers. Experimental results show that this approximation only slightly degrades compression performance, and it greatly simplifies the generation of adaptive quantization tables. To compress a new image, we just need an efficient classification network to identify its category, and then we use the corresponding center quantization table as its adaptive

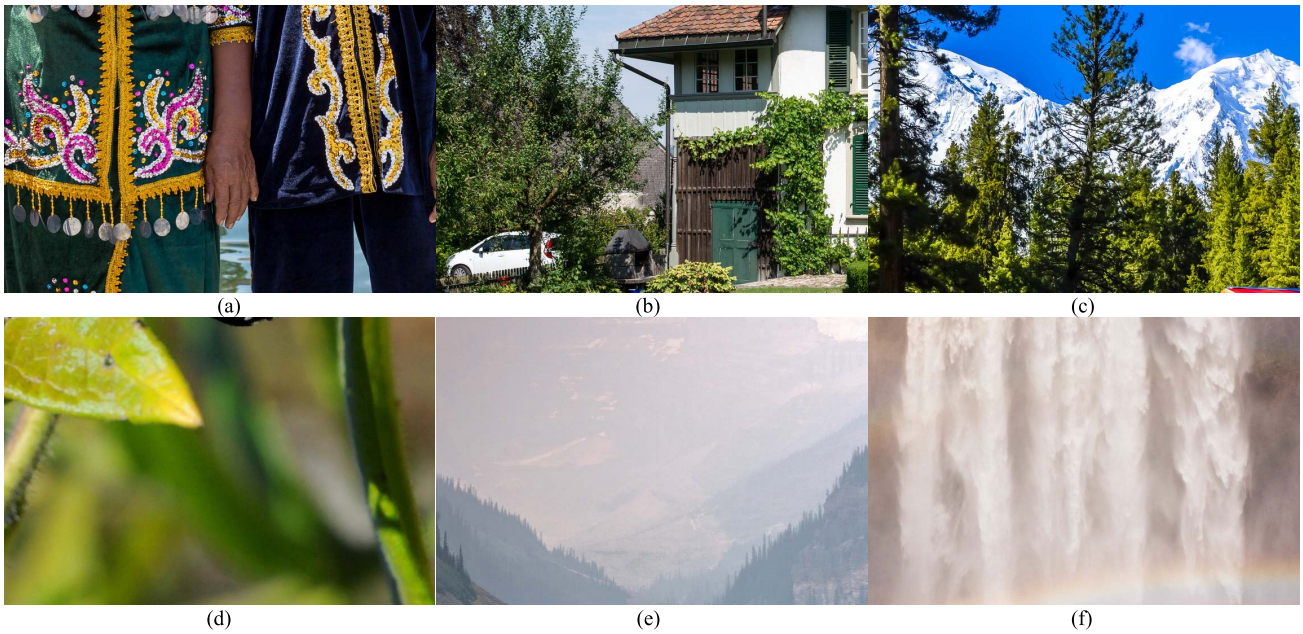


Fig. 4. Images with different frequency domain features. (a) - (c) are rich in detail and (d) - (f) are relatively flat.

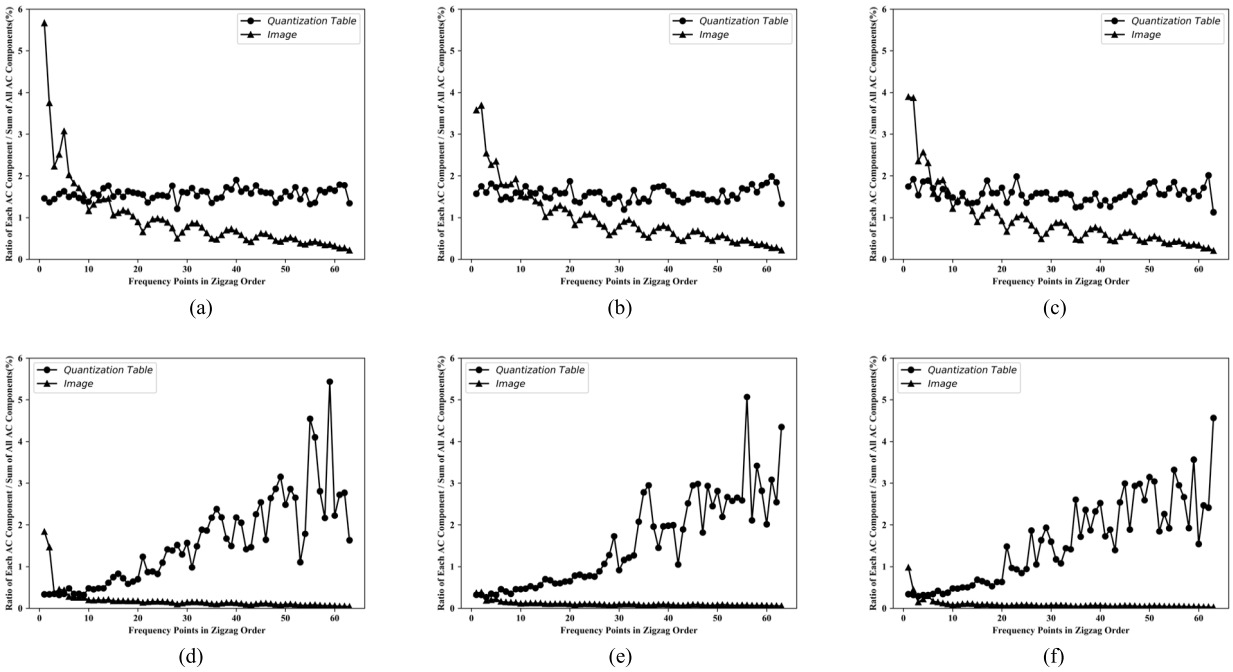


Fig. 5. Distribution of the AC coefficients and optimal quantization tables for images in Fig. 4. (a)-(f) correspond to Figs. 4 (a)-(f), respectively. The horizontal axis represents the frequency points in zigzag order. For images, the vertical axis represents the ratio of the sum of coefficients at each single AC frequency point to the sum of all AC coefficients, and for quantization tables the vertical axis represents the ratio of each AC entry to the sum of all AC entries. We can see that images with similar frequency domain distributions, such as (a)-(c) and (d)-(f), have very similar quantization tables.

quantization table. K-means clustering aims to partition N observations into K clusters, where each observation belongs to the cluster with the smallest Euclidean distance. To directly use K-means to classify the quantization table is susceptible to isolated points and noise. The PCA algorithm is a way to reduce the dimensions of data while minimizing information loss. Here, we first project the 64-dimensional quantization table data into 48-dimensional space through PCA, and then use K-Means clustering to place the resulting 48-dimensional vectors into several categories.

D. QNet-C: Classification Network

After obtaining several cluster center quantization tables, the goal of the network is just to identify the right one for each input image, which can be regarded as a typical classification task. Since the images have been labeled as the category to

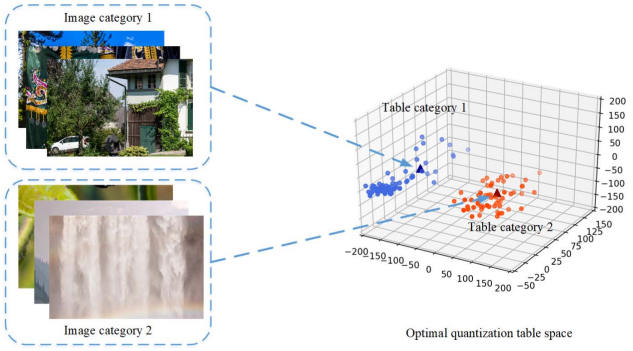


Fig. 6. The optimal quantization tables are placed into two categories by the PCA and K-Means algorithms and the images are also classified according to the category to which its optimal quantization table belongs. Then the cluster center quantization tables (represented by triangles) are used as the adaptive quantization tables for images in each category. This figure chooses some typical quantization tables and project them into 3D space through PCA for clustering. In practice, K-Means clustering is performed in 48-dimensional space.

which its optimal quantization table belongs, the classification task can be established naturally. The number of image categories is usually much smaller than 64, hence, QNet-C can be greatly simplified compared to QNet-R. As shown in Fig. 7, we remove the inception module and halve the output channels of the 1×1 convolution. The number of parameters in the fully connected layer is also greatly reduced.

IV. Experimental Results

A. Training Dataset Setup

DIV2K is a high-resolution image set of the NTIRE 2017 challenge on single image super resolution, including 800 training images, 100 validation images, and 100 test images [24]. For this paper, we obtained 16,000 images by mirroring and cropping on the training set of DIV2K, and then we calculated the optimal quantization table corresponding to each image through the classical genetic algorithm to establish a training dataset. The validation and test set were created similarly, and each contained 2000 images. Finally, the comparison between the proposed method and previous methods was made on the test set, the Kodak image set, and an image set provided by Alibaba. The widely used Kodak image set contains 24 natural images, while the Alibaba image set contains 100 artificially processed images, including text, faces, and commodities. Experiments were conducted on the set of standard 8-bit grayscale images with a resolution of 768×512.

There are many types of research of the generation of self-adaptive optimal quantization tables, including rate-distortion optimization [11], the genetic algorithm [15], and the simulated annealing algorithm [16]. The method proposed in this paper is applicable to the optimal quantization table dataset generated by any algorithm. Since it does not affect the computational complexity of the inference phase, we used the classical genetic algorithm with large computational complexity to ensure the accuracy of the dataset.

The genetic algorithm is a kind of heuristic algorithm. Its main idea is to generate a large number of individuals through crossover and mutation methods based on the previous generation of data and then to evaluate each individual's unfitness index and retain a certain number of outstanding individuals as parents, repeating this process until the smallest unfitness index no longer changes. In this experiment, the whole genetic algorithm aims to find the quantization table with the smallest unfitness index for each image. Without loss of generality, the mutation method is defined as randomly adjusting a random item of one quantization table within $\pm 10\%$ and the crossover method is defined as exchanging a random item of two quantization tables. The unfitness index is used to characterize the degree of unsuitability of each quantization table. We established training sets under both PSNR and SSIM to prove that the method proposed in this paper is also independent of the image quality assessment. For simplicity, the following description takes PSNR as an example, and operations are the same for SSIM. Let S^* and P^*, respectively, be the compressed file size and PSNR value of the baseline compression using the standard quantization table. Similarly, let S_i and P_i be the compressed file size and PSNR value of the compression using the newly generated table T_i in the genetic algorithm. Since the genetic algorithm tries to find the quantization table with the highest image quality and no compression rate degradation, the difference of file size is magnified by a factor of 10 as a penalty when $S_i > S^*$, so as to eliminate quantization tables that do not meet the compression ratio requirement. When $S_i \leq S^*$, the unfitness index must only consider the change in image quality. The unfitness index of quantization table T_i is

$$U(T_i) = \begin{cases} (S_i - S^*) \times 10 + (P^* - P_i), & if\ S_i > S^* \\ P^* - P_i, & otherwise. \end{cases} \quad (1)$$

For each image, we first use the mutation method to generate 500 new quantization tables based on the standard quantization table, and then evaluate the unfitness index of each quantization table, retaining the top 50 quantization tables. From the second round on, each iteration utilizes the mutation and crossover methods to generate 500 new quantization tables based on the 50 quantization tables that survived the previous iterations, evaluates their unfitness index and retains the top 50 of the 550 quantization tables (the 50 tables retained by last iteration and the 500 tables generated by this iteration). When the minimum unfitness index no longer shrinks during 10 consecutive iterations, the genetic algorithm is considered to reach convergence, and the quantization table corresponding to the minimum unfitness index is taken as the optimal quantization table for the image.

B. Representative Quantization Tables

In practical applications, the number of clusters can be determined according to the diversity of image features. For the previously mentioned dataset, we tried to divide the optimal quantization table data into 15, 10, 5, and 3 categories and use the cluster center quantization table to compress the images in each category. The original optimal quantization tables offer an average PSNR improvement of 1.72 dB at the same compression rate of standard table with a quality level of 80, and that of the cluster center quantization tables corresponding to the 15, 10, 5, 3 categories are 1.70, 1.69,

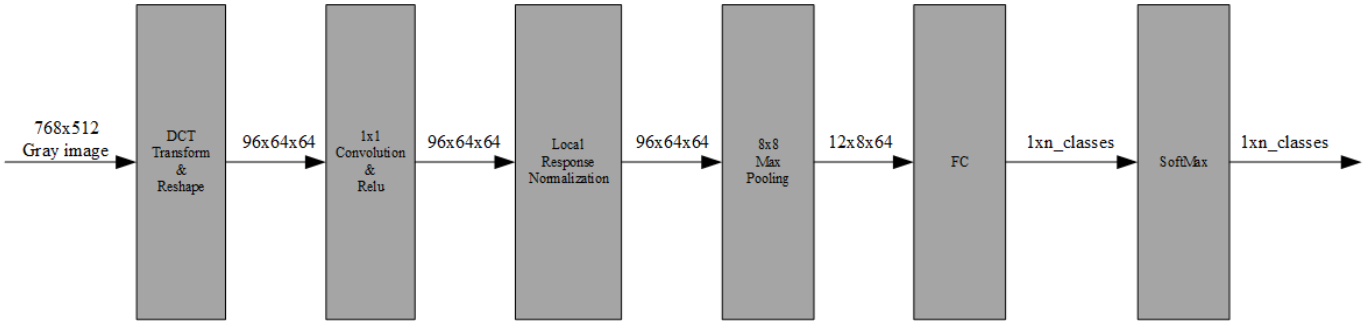


Fig. 7. The structure of the classification network.

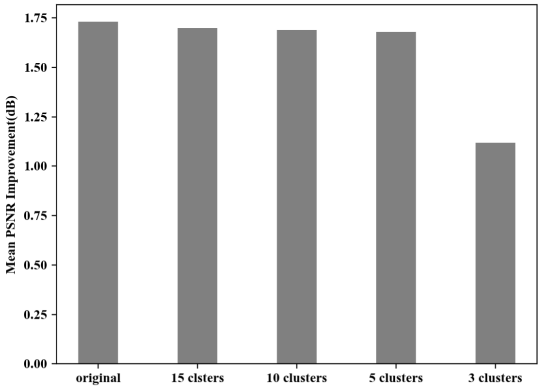


Fig. 8. The mean PSNR improvement of different clustering number. Result of the original optimal quantization tables is also illustrated.

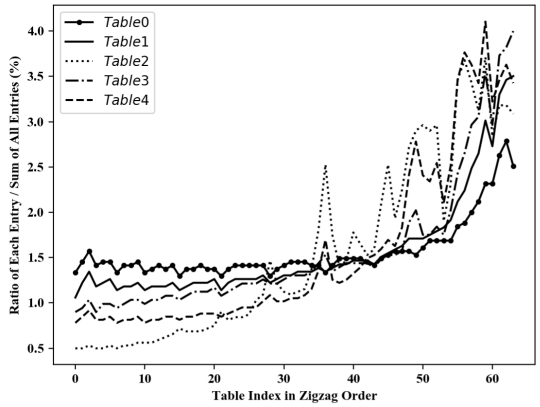


Fig. 9. The distribution of the extracted quantization tables. The horizontal axis represents the table index in the zigzag order and the vertical axis represents the ratio of each entry to the sum of all entries.

1.67, 1.12dB. As shown in Fig. 8, the PSNR gain does not significantly degrade until the tables are placed into three categories. Considering both the simplification and PSNR improvement, we select the case of five clusters as learning target. The frequency domain distribution of the five cluster center quantization tables is shown in Fig. 9. Essentially, the quantization table defines the proportion of quantization loss at each frequency point. It can be clearly seen that the five optimal quantization tables represent five kinds of loss distribution in the DCT domain, which further demonstrates that the optimal quantization table is closely related to the frequency domain feature of an image.

C. CNN Training

In experiments, we found that the network using only AC coefficients as input can reduce the amount of computation while accelerating the convergence, which implies that the quantization table is mainly dependent on the AC coefficient distribution of the image. Therefore, we convert the 768×512 image to a $96 \times 64 \times 63$ tensor as the input of the network, with the last dimension representing 63 AC coefficients.

In order to improve the generalization ability and stability of the network, we also need to enhance the training set. Since the JPEG standard is to compress images in units of 8×8 blocks, we increase the data space by randomly rearranging 8×8 blocks. Experiments show that the optimal tables still have the same benefits on rearranged images. This also implies that the neural network extracts statistical features rather than the local relationship in the spatial domain.

By analyzing the tables, we found that the high-frequency entries are generally significantly larger than the low-frequency entries, but the latter has a greater influence on compression performance. Thus, we scale the ground truth tables according to the magnitude of each entry in the standard table, so as to make the contribution of each entry on the loss function fairer. In the inference phase, the optimal quantization table could be obtained by reverse scaling the output of the regression network. Denote the standard quantization table as T^*. Then the scaling factor is defined as

$$F_i = \frac{\min_{0 \le j \le 63} T_j^*}{T_i^*}, \quad 0 \le i \le 63. \tag{2}$$

The MSE between the network output and the scaled quantization table is used as the loss function. To avoid overfitting, we add L2 regularization to the loss function. The hyper-parameter α is employed to adjust the contribution of the L2 regularization and MSE to the final loss. Let f_R be the transformation function of the regression network. Then the final loss function can be written as

$$L_R(\omega, x) = MSE\left(f_R(\omega, x), T'_{opt}\right) + \alpha L_2(\omega). \tag{3}$$

where x is the AC coefficient matrix, ω is the coefficient of the regression network, and T'_{opt} is the scaled optimal quantization table.

For the classification task, the category index of each image is converted to a one-hot vector v to train the network. The softmax function S and cross-entropy function H are used

TABLE I

PSNR COMPARISON OF DIFFERENT Q-TABLE OPTIMIZATION METHODS FOR 512 × 512 LENA

Rate	Baseline	OptD-HDQ	Anneal	QNet-R	QNet-C
0.50	34.90	35.53	35.96	35.56	**36.20**
0.75	36.62	37.77	38.22	37.80	**38.43**
1.00	37.91	39.31	39.68	39.31	**39.89**
1.25	38.98	40.52	40.87	40.39	**40.92**
1.50	39.96	41.69	41.81	41.49	**41.90**
1.75	40.75	42.71	42.70	42.52	**42.81**
2.00	41.66	43.84	43.67	43.49	**43.90**

TABLE II

PSNR COMPARISON OF DIFFERENT Q-TABLE OPTIMIZATION METHODS FOR 512 × 512 GOLDHILL

Rate	Baseline	OptD-HDQ	Anneal	QNet-R	QNet-C
0.50	31.72	32.30	32.34	32.40	**32.51**
0.75	33.26	34.25	34.28	34.45	**34.50**
1.00	34.55	35.88	35.79	36.03	**36.12**
1.25	35.62	37.27	37.10	37.37	**37.50**
1.50	36.65	38.55	38.30	38.50	**38.73**
1.75	37.66	39.73	39.50	39.55	**39.89**
2.00	38.53	40.93	40.54	40.55	**40.98**

in the loss function, and L2 regularization is added to avoid overfitting. Similar to (3), the loss function of the classification network is

$$L_C(\omega, x) = H\left(S(f_C(\omega, x)), \upsilon\right) + \alpha L_2(\omega). \tag{4}$$

The regression and classification networks are trained using the TensorFlow framework and the Adam optimizer [27].

D. Comparison Results

We first compare the results of QNet-R and QNet-C to Yang *et al.* [11] and Max *et al.* [16] under PSNR. To facilitate our subsequent discussion, we shall refer to compression with quantization table generated by Yang *et al.* [11] and Max *et al.* [16] as OptD-HDQ and Anneal. As mentioned earlier, the OptD-HDQ and Anneal represent the state-of-the-art JPEG quantizers obtained by rate-distortion optimization and heuristic algorithms. Since Anneal produces several optimal quantization tables, we compress the images using each optimal table and select the best result as a comparison. For OptD-HDQ, we just quote the results of hard decision quantization claimed in the paper. The performance of baseline JPEG encoder using the standard quantization table is shown as an anchor. Tables I and II show the PSNR performance of these optimization methods for 512 × 512 Lena and Glodhill.

Figs. 10-12 show the mean PSNR of the test set, Kodak image set, and Alibaba image set, respectively. Each is compressed using the quantization table obtained by Anneal, QNet-R, and QNet-C. It can be clearly seen that our proposed method achieves greater improvements. The R-D curve of QNet-R is slightly higher than that of Anneal, and QNet-C performs significantly better than the other two methods when the compression rate is greater than 0.75, which is most common. At the compression rate of 1.0 bpp, QNet-R and QNet-C provide average PSNR gains of nearly 1.2 and 1.4 dB on the test set, Kodak image set, and Alibaba image set, compared to the standard table.

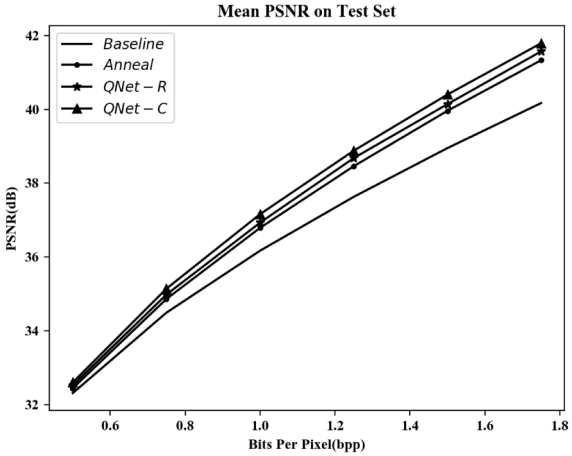


Fig. 10. Mean PSNR of the 2000 images in test set.

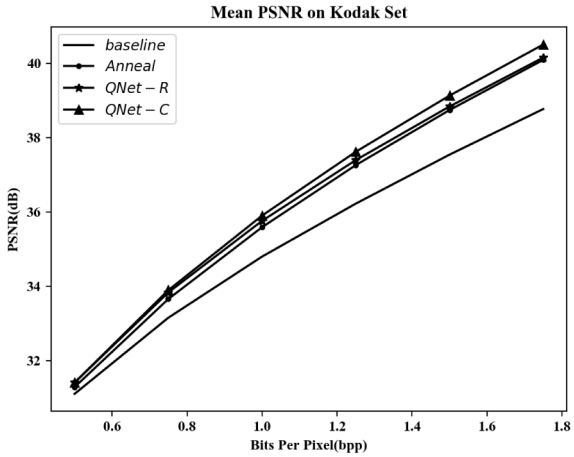


Fig. 11. Mean PSNR of the 24 images in Kodak image set.

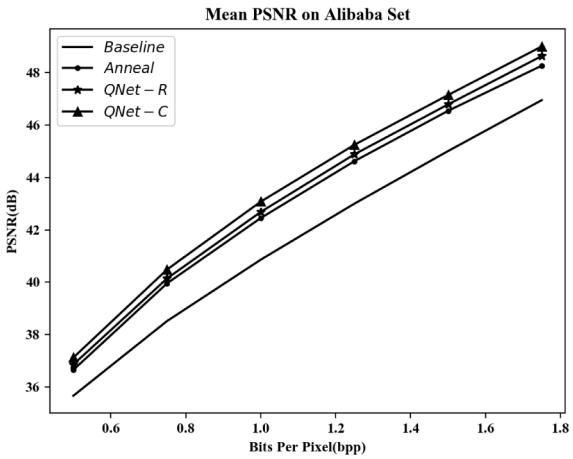
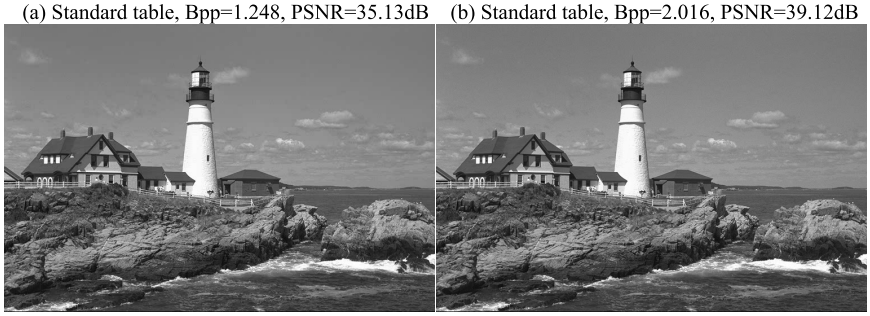


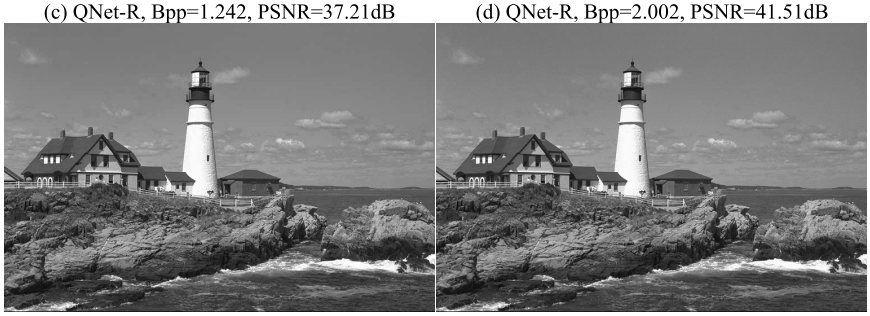
Fig. 12. Mean PSNR of the 100 images in Alibaba image set.

Fig. 13 shows the subjective quality comparison on Kodak Image 21. Since most applications usually require high image quality, we compare the images under compression ratios of

(a) Standard table, Bpp=1.248, PSNR=35.13dB (b) Standard table, Bpp=2.016, PSNR=39.12dB



(c) QNet-R, Bpp=1.242, PSNR=37.21dB (d) QNet-R, Bpp=2.002, PSNR=41.51dB



(e) QNet-C, Bpp=1.253, PSNR=37.34dB (f) QNet-C, Bpp=1.964, PSNR=41.38dB

Fig. 13. The subjective quality comparison on Kodak Image 21. The compression ratios of left and right column are target at 1.25 and 2.0 bpp respectively. The top row is compressed by using the standard table while the middle and bottom rows are the compression results with the adaptive quantization tables offered by QNet-R and QNet-C.

TABLE III

SSIM COMPARISON OF DIFFERENT Q-TABLE OPTIMIZATION METHODS FOR 768×512 KODIM01

Rate	Baseline	QNet-R	QNet-C	QNet-R Profit	QNet-C Profit
0.50	0.7721	0.7723	0.7726	0.03%	0.06%
0.75	0.8398	0.8409	0.8440	0.13%	0.50%
1.00	0.8821	0.8861	0.8879	0.45%	0.66%
1.25	0.9095	0.9170	0.9181	0.82%	0.95%
1.50	0.9288	0.9381	0.9392	1.00%	1.12%
1.75	0.9429	0.9531	0.9542	1.08%	1.20%
2.00	0.9545	0.9639	0.9651	0.98%	1.11%

TABLE IV

SSIM COMPARISON OF DIFFERENT Q-TABLE OPTIMIZATION METHODS FOR 768×512 KODIM02

Rate	Baseline	QNet-R	QNet-C	QNet-R Profit	QNet-C Profit
0.50	0.8761	0.8765	0.8769	0.05%	0.09%
0.75	0.9104	0.9136	0.9151	0.35%	0.52%
1.00	0.9320	0.9366	0.9372	0.49%	0.56%
1.25	0.9459	0.9515	0.9527	0.59%	0.72%
1.50	0.9569	0.9621	0.9637	0.54%	0.71%
1.75	0.9649	0.9702	0.9718	0.55%	0.72%
2.00	0.9719	0.9762	0.9780	0.44%	0.63%

1.25 and 2.0 bpp. It can be seen that the textures of the images compressed with the adaptive quantization tables offered by QNet-R and QNet-C are better preserved.

To illustrate that the proposed method is applicable to various image quality assessments, we trained networks under the SSIM dataset. Tables III–V show the SSIM performance of the classification and regression networks for Kodim 01-03. It is seen from the table that the SSIM index has been improved significantly over different compression ratios.

In addition to image quality, the computational cost is an important factor on which to evaluate optimization algorithms. Since Anneal aims to produce an optimal universal quantization table, it does not require additional computation during image compression. Other methods [10], [11], [15] are mostly based on iterative algorithms, which usually result in a large computational workload. Manavalan *et al.* [15] usually takes about 100 iterations to obtain the optimal quantization table, requiring more than 2500 s to process a 768×512 image on a single core at 3.20 GHz. Rate-distortion optimization has higher computational efficiency than heuristic algorithms. The methods proposed in V. Ratnakar and M. Livny [10] and Yang *et al.* [11] require 1.95 s and 0.9 ms, respectively, to process a 768×512 image on an Apple Mac Pro 8-core 2.4 GHz computer. In contrast, it takes 15 and

TABLE V
SSIM COMPARISON OF DIFFERENT Q-TABLE OPTIMIZATION METHODS FOR 768 × 512 KODIM03

Rate	Baseline	QNet-R	QNet-C	QNet-R Profit	QNet-C Profit
0.50	0.9309	0.9315	0.9320	0.06%	0.12%
0.75	0.9565	0.9590	0.9600	0.26%	0.37%
1.00	0.9695	0.9720	0.9734	0.26%	0.40%
1.25	0.9769	0.9792	0.9804	0.24%	0.36%
1.50	0.9817	0.9837	0.9846	0.20%	0.30%
1.75	0.9851	0.9865	0.9873	0.14%	0.22%
2.00	0.9873	0.9890	0.9895	0.17%	0.22%

6.25ms, respectively, for QNet-R and QNet-C to process a 768 × 512 image on a single CPU core at 3.20 GHz. For comparison, the vanilla JPEG encoder needs about 95 ms under the same conditions. Thus, the CNN-based quantization table generator proposed in this paper is easily added to the JPEG compression process with no significant performance degradation.

V. CONCLUSION

We proposed the CNN-based method QNet to adaptively generate an optimal quantization table in a standard-compliant way. Compared to other quantization table optimization methods, QNet has the advantages of image quality assessment independence and high efficiency. The 1 × 1 convolution and inception module were used to extract and fuse features in the frequency and spatial domains without confusion. The regression network was trained on a dataset containing more than 10,000 images and their corresponding optimal quantization tables. In addition, we adopted PCA and K-means clustering to place the images and optimal quantization tables into several categories and trained a classification network to further improve the compression performance and computational efficiency. We evaluated the compression results on the test set, Kodak image set, and Alibaba image set, and results showed that the regression network and classification network gained significant benefits under both PSNR and SSIM compared to the standard table provided by the JPEG standard. At the compression rate of 1.0 bpp, QNet-R and QNet-C provided average PSNR gains of nearly 1.2 and 1.4 dB on the test set, Kodak image set, and Alibaba image set. For the experiment under SSIM, the improvements were 0.4% and 0.54%, respectively. In terms of performance, the regression and classification network only took 15 and 6.25 ms, respectively, to process a 768 × 512 image on a single CPU core at 3.20 GHz. In summary, QNet can achieve significant improvements in compression performance with little computational workload.

REFERENCES

[1] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Trans. Consum. Electron.*, vol. 38, no. 1, pp. xviii–xxxiv, Feb. 1992.

[2] E.-H. Yang and L. Wang, "Joint optimization of run-length coding, Huffman coding, and quantization table with complete baseline JPEG decoder compatibility," *IEEE Trans. Image Process.*, vol. 18, no. 1, pp. 63–74, Jan. 2009.

[3] R. A. V. Kam, P. W. Wong, and R. M. Gray, "JPEG-compliant perceptual coding for a Grayscale image printing pipeline," *IEEE Trans. Image Process.*, vol. 8, no. 1, pp. 1–14, Jan. 1999.

[4] G. Lakhani, "Modified JPEG Huffman coding," *IEEE Trans. Image Process.*, vol. 12, no. 2, pp. 159–169, Feb. 2003.

[5] G. Lakhani and V. Ayyagari, "Improved Huffman code tables for JPEG's encoder," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, no. 6, pp. 562–564, Dec. 1995.

[6] G. Lakhani, "A modification to the Huffman coding of JPEG's baseline compression algorithm," in *Proc. DCC. Data Compress. Conf.*, Snowbird, UT, USA, Mar. 2000, p. 557.

[7] J. Huang and T. Meng, "Optimal quantizer step sizes for transform coders," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Apr. 1991, pp. 2621–2624.

[8] S. W. Wu and A. Gersho, "Rate-constrained picture-adaptive quantization for JPEG baseline coders," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Apr. 1993, pp. 389–392.

[9] M. Crouse and K. Ramchandran, "Joint thresholding and quantizer selection for transform image coding: Entropy-constrained analysis and applications to baseline JPEG," *IEEE Trans. Image Process.*, vol. 6, no. 2, pp. 285–297, Feb. 1997.

[10] V. Ratnakar and M. Livny, "An efficient algorithm for optimizing DCT quantization," *IEEE Trans. Image Process.*, vol. 9, no. 2, pp. 267–270, Feb. 2000.

[11] E. Yang, C. Sun, and J. Meng, "Quantization table design revisited for image/video coding," *IEEE Trans. Image Process.*, vol. 23, no. 11, pp. 4799–4811, Nov. 2014.

[12] L.-W. Chang, C.-Y. Wang, and S.-M. Lee, "Designing JPEG quantization tables based on human visual system," in *Proc. Int. Conf. Image Process.*, Kobe, Japan, vol. 2, Oct. 1999, pp. 376–380.

[13] Y. Jiang and M. S. Pattichis, "JPEG image compression using quantization table optimization based on perceptual image quality assessment," in *Proc. Conf. Rec. 45th Asilomar Conf. Signals, Syst. Comput. (ASILOMAR)*, Pacific Grove, CA, USA, Nov. 2011, pp. 225–229.

[14] X. Zhang, S. Wang, K. Gu, W. Lin, S. Ma, and W. Gao, "Just-noticeable difference-based perceptual optimization for JPEG compression," *IEEE Signal Process. Lett.*, vol. 24, no. 1, pp. 96–100, Jan. 2017.

[15] V. K. Balasubramanian and K. Manavalan, "Knowledge-based genetic algorithm approach to quantization table generation for the JPEG baseline algorithm," *TURKISH J. Electr. Eng. Comput. Sci.*, vol. 24, pp. 1615–1635, Mar. 2016.

[16] M. Hopkins, M. Mitzenmacher, and S. Wagner-Carena, "Simulated annealing for JPEG quantization," in *Proc. Data Compress. Conf.*, Snowbird, UT, USA, Mar. 2018, p. 412.

[17] K. Ramchandran and M. Vetterli, "Rate-distortion optimal fast thresholding with complete JPEG/MPEG decoder compatibility," *IEEE Trans. Image Process.*, vol. 3, no. 5, pp. 700–704, Sep. 1994.

[18] J. Alakuijala, R. Obryk, O. Stoliarchuk, Z. Szabadka, L. Vandevenne, and J. Wassenberg, "Guetzli: Perceptually guided JPEG encoder," 2017, *arXiv:1703.04421*. [Online]. Available: http://arxiv.org/abs/1703.04421

[19] G. Toderici *et al.*, "Full resolution image compression with recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 5435–5443.

[20] L. Theis, W. Shi, A. Cunningham, and F. Huszár, "Lossy image compression with compressive autoencoders," 2017, *arXiv:1703.00395*. [Online]. Available: http://arxiv.org/abs/1703.00395

[21] O. Rippel and L. Bourdev, "Real-time adaptive image compression," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, 2017, pp. 2922–2930.

[22] M. Lin, Q. Chen, and S. C. Yan, "Network in network," 2013, *arXiv:1312.4400*. [Online]. Available:https://arxiv.org/abs/1312.4400

[23] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1–9.

[24] E. Agustsson and R. Timofte, "NTIRE 2017 challenge on single image super-resolution: Dataset and study," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Honolulu, HI, USA, Jul. 2017, pp. 1122–1131.

[25] Z. Liu *et al.*, "DeepN-JPEG: A deep neural network favorable JPEG-based image compression framework," in *Proc. 55th ACM/ESDA/IEEE Design Autom. Conf. (DAC)*, San Francisco, CA, USA, Jun. 2018, pp. 1–6.

[26] J. Ballé, V. Laparra, and E. P. Simoncelli, "End-to-end optimized image compression," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–27.

[27] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," presented at the 3rd Int. Conf. for Learn. Represent., 2014.

Xiao Yan received the B.S. and M.S. degrees in microelectronics from Xidian University, Xian, China, in 2011 and 2014, respectively. He is currently pursuing the Ph.D. degree with the College of Microelectronics, Fudan University.

From 2014 to 2018, he was a Research Staff Member with Huawei Technologies, Xian, China. His research interests mainly include image processing and associated VLSI architecture.

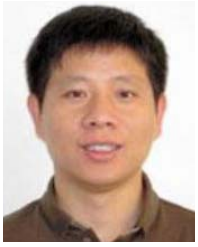
Yibo Fan (Member, IEEE) received the B.E. degree in electronics and engineering from Zhejiang University, Hangzhou, China, in 2003, the M.S degree in microelectronics from Fudan University, Shanghai, China, in 2006, and the Ph.D. degree in engineering from Waseda University, Tokyo, Japan, in 2009. He was an Assistant Professor with Shanghai Jiao Tong University and Fudan University from 2009 to 2014, and Associate Professor with Fudan University from 2014 to 2019. He is currently a Full Professor with the College of Microelectronics, Fudan University. His research interests include image processing, video coding, machine learning and associated VLSI architecture.

Kewei Chen received the B.S. degree in microelectronics from Fudan University, Shanghai, China, in 2017, where he is currently pursuing the M.S. degree in microelectronics.

His research interests include deep learning and image processing algorithms.

Xulin Yu received the master's degree from Zhejiang University, Hangzhou, China, in 2010.

He is currently the Director of heterogeneous computing with Alibaba Infrastructure Service. His research interests include heterogeneous computing, algorithms and VLSI architectures for deep learning and multimedia signal processing.



Xiaoyang Zeng (Member, IEEE) received the B.S. degree from Xiangtan University, Xiangtan, China, in 1992, and the Ph.D. degree from the Changchun Institute of Optics, Fine Mechanics, and Physics, Chinese Academy of Sciences, Changchun, China, in 2001.

From 2001 to 2003, he was a Postdoctoral Researcher with Fudan University, Shanghai, China. Then, he joined the State Key Lab of ASIC and System, Fudan University, as an Associate Professor, where he is currently a Full Professor and the Director. His research interests include information security chip design, system-on-chip platforms, and VLSI implementation of digital signal processing and communication systems.