Multiple Description Convolutional Neural Networks for Image Compression

Lijun Zhao, Huihui Bai, Member, IEEE, Anhong Wang, Member, IEEE, and Yao Zhao, Senior Member, IEEE

Abstract—Multiple description coding (MDC) is able to stably transmit the signal in the un-reliable and non-prioritized networks, which has been broadly studied for several decades. However, the traditional MDC doesn’t well leverage image’s context features to generate multiple descriptions. In this paper, we propose a novel standard-compliant convolutional neural network-based MDC framework in term of image’s context features. Firstly, multiple description generator network (MDGN) is designed to produce appearance-similar yet feature-different multiple descriptions automatically according to image’s content, which are compressed by standard codec. Secondly, we present multiple description reconstruction network (MDRN) including side reconstruction networks (SRN) and central reconstruction network (CRN). When any one of two lossy descriptions is received at the decoder, SRN network is used to improve the quality of this decoded lossy description by removing the compression artifact and up-sampling simultaneously. Meanwhile, we utilize CRN network with two decoded descriptions as inputs for better reconstruction, if both of lossy descriptions are available. Thirdly, multiple description virtual codec network (MDVCN) is proposed to bridge the gap between MDGN network and MDRN network in order to train an end-to-end MDC framework. Here, two learning algorithms are provided to train our whole framework. In addition to structural similarity loss function, the produced descriptions are used as opposing labels with multiple description distance loss function to regularize the training of MDGN network. These losses guarantee that the generated description images are structurally similar yet finely diverse. Experimental results show a great deal of objective and subjective quality improvements in terms of appropriate construction of lattice vector quantization and different subbands quantization step on the wavelet domain. In [8], the index assignment of multiple description description lattice vector quantization is operated in an optimized way in terms of appropriate construction of wavelet coefficient vectors, choosing sub-lattice index values and different subbands quantization step on the wavelet domain. In [3], the index assignment of multiple description lattice vector quantization is designed to be translated into a transportation problem and greedy algorithm as well as general algorithms is developed to pursue optimality of the index assignment.

I. INTRODUCTION

LARGE amounts of attentions have been paid to various techniques of Internet service and multimedia signal transmission for many years, which not only provide us a convenient manner of communication but also give us many choices for our life style. Meanwhile, the bandwidth of Internet has been accelerated and more stable transmission service is guaranteed by these developments. But there are still some risks of transmission failures, when the Internet congestion occurs in the overloaded case or signal packets are conveyed in the unpredictable yet unreliable channels. Multiple description coding has been studied as a promising technique of source coding to relieve these problems by decomposing the signal into multiple redundant subsets, which are transmitted in different channels. Thus, a degraded but acceptable signals reconstruction can be produced after decoding, even though only one description is received at the clients. If more descriptions are available for users, better quality of signal reconstruction can be achieved.

As one of the main techniques in multiple description image coding, multiple description scalar quantization could overcome impairments of transmission channel [6]. For example, in [7], multiple description scalar quantizers have been combined with efficient wavelet coders to generate independent multiple packets for error resilience. In [5], two-stage multiple description scalar quantization is presented to create central and side decoders, whose distortions are closer to the rate-distortion bound of multiple description coding under the condition of the high-resolution assumption. To cope with the L-description problem [9], two novel coding schemes are proposed, when the symmetric rates and symmetric distortion is constrained. In [8], a new achievable rate-distortion region with combinatorial message sharing is presented by introducing shared codebooks and the refinement codebook to generate L-channel multiple descriptions.

Compared with multiple description scalar quantization, lattice vector quantization characterizes in good symmetric structure of lattices and avoiding complex nearest neighbor searching. In [10], the main problem of designing lattice vector quantizer is formulated as a labeling problem for two-channel multiple description. In [11], non-lattice codebook with symmetries of the coarse lattice is used to get objective quality gains for multiple description coding but without a great increase of complexity. In [12], multiple description lattice vector quantization is operated in an optimized way in terms of appropriate construction of wavelet coefficient vectors, choosing sub-lattice index values and different subbands quantization step on the wavelet domain. In [3], the index assignment of multiple description lattice vector quantization is designed to be translated into a transportation problem and greedy algorithm as well as general algorithms is developed to pursue optimality of the index assignment.

Except multiple descriptions directly produced by quantization, there are many alternative strategies for multiple description coding. To generate two descriptions in...
transform based coding framework, correlation between pairs of transform coefficients is introduced by a pairwise correlating transform [13]. This correlation facilities to reduce the distortion when only a single description is received. Later, both domain-based multiple description coding and forward error correction are used for concatenated multiple description coding of frame-rate scalable video [14]. Meanwhile, both prioritized discrete cosine transform in video compression and multiple description codes based on forward error correction are combined together to provide a wireless channels video transmission scheme [15].

From literatures [14,15], it can be observed that multiple description video coding using forward error correction has been widely explored. There are several other kinds of multiple description video coding. In [16], a video is coded into multiple independently streams so that each stream has its own prediction and dependent state to defeat against bit error or packet loss. In multiple description motion coding algorithm, motion vector is encoded into two descriptions, which are transmitted over distinct channels to the decoder so that motion vector field is robust against transmission errors [17]. In the scalable wavelet video codec, each packet is encoded with a separate channel code, so that the integrity of the packets is protected and it allows to detect packet-decoding failures cases, after breaking wavelet transformation into several spatial-temporal tree blocks [18]. In [19], two architectures of multiple description video coding are built up based on motion compensation prediction loop and a poly-phase down-sampling technique is chosen to generate multiple descriptions and introduce cross redundancy among the descriptions.

Although the aforementioned approaches can well alleviate the congestion of Internet and satisfy the demanding of real-time application, these approaches are not compatible to standard codec, such as JPEG, and JPEG2000. To resolve this problem, some previous works have provided some feasible solutions, such as [4, 8, 20, 21]. In [21], through grouping the codeblock to generate two balanced set, these two set are compressed by JPEG-2000 with two different quantization parameter to get four subsets, which are interlacedly merged together to create two descriptions. In [20], the rate-allocation strategy embedded in the JPEG2000 encoder is introduced for the rate-distortion optimization of multiple descriptions of images, in which single description decoding is able to compatible with JPEG2000 Part 1 decoder. In view of human eyes’ always sensitivity to the changes above just noticeable difference (JND) threshold, only the significant visual information, which contributes to the JND tolerance, is encoded as the redundant information during H.264/AVC based multiple description video coding [4]. In [8], frame-level rate-distortion optimized description generation scheme takes account of temporal coding dependency to minimize the end-to-end distortion, which is built on standard H.264/AVC.

Because the proposed approach is high related about the issue of compression artifact removal [22–30], we will next review several state-of-the-art works about compression artifact removal. In [22], pointwise shape-adaptive discrete cosine transform is leveraged for both denoising and deblocking after image compression. In [23], dictionary learning is introduced to reducing JPEG-compressed artifacts in view of image’s sparse and redundant representations. In [24], collaborative filtering is designed to uncover the finest details and maintain each individual block’s unique features in the sparse 3-D transform-domain, which is not restrict to the denoising of compressed image, so this approach is a general denoising method. Lately, the deblocking problem is formulated as an optimization problem, where non-convex low-rank model constrained is considered to reduce blocking artifacts [25]. Meanwhile, the popular techniques of convolutional neural network and generative adversarial have been tried to remove artifacts [27,29].

Following the work of [19], we form multiple description coding baselines with a poly-phase down-sampling technique to generate multiple descriptions by combining state-of-the-art artifact removal technique with super-resolution based on very deep convolutional neural network. Specifically, the input image is down-sampled with a poly-phase down-sampling technique along the main diagonal for each 2 × 2 non-overlapped window to form two descriptions for coding with standard codec. After decoding, several state-of-the-art artifact removal techniques, such as [22,25] are used to enhance image quality, which is followed by super-resolution to restore image from low-resolution to high-resolution with very deep convolutional neural network, such as novel super-resolution methods of [31] and [32]. The combinatorial methods with artifact removal [22,25] and super-resolution [31] are respectively referred to as multiple description coding baselines 1-4, namely “MDB1a”, “MDB2a”, “MDB3a”, “MDB4a”. In this similar way, when artifact removal methods of [22,25] are combined together with [32], they are respectively denoted as “MDB1b”, “MDB2b”, “MDB3b”, “MDB4b”.

In this paper, we introduce a novel standard-compatible multiple description coding framework, in which multiple descriptions are produced by deep convolutional neural network. Our contributions are listed as follows:

- Multiple description generator network (MDGN) is introduced to adaptively generate multiple descriptions according to image’s content, which are compressed by standard codec to reduce transmission bits.
- We present multiple description reconstruction network (MDRN), which consists of side reconstruction networks (SRN) and central reconstruction network (CRN). When either one of two compressed description is received at the decoder, side reconstruction network-A network (SRNA) or side reconstruction network-B (SRNB) is used to reconstruct the losty description and enlarge this description simultaneously by removing compression artifact and up-sampling. Meanwhile, we utilize CRN network with two received descriptions as inputs to achieve high-quality image reconstruction, if all the multiple description images are available.
We train the aforementioned two neural networks: MDGN network and MDRN network together by learning multiple description virtual codec network (MDVCN). It means that the learned MDVCN network is leveraged to further supervise the MDGN network's training. Besides, we provide two kinds of learning algorithms for training our convolutional neural networks.

- Distance loss for MDGN network is introduced as well as structural similarity loss to guarantee that the generated description images are structurally similar yet finely different.

The rest of this paper is given as follow. We first introduce the proposed methodology in Section 2. After that, we conduct a series of the experimental results to validate the efficiency in the Section 3. At last, we give a conclusion in the Section 4.

II. THE METHODOLOGY

In this paper, multiple description coding framework based on deep convolutional neural network is introduced to efficiently compress images, when facing an unpredictable and non-prioritized channel. Our main works are put on how to generate multiple descriptions in terms of redundancy between each description and description’s diversity for better central reconstruction. Meanwhile, we design the neural network for description’s generation and reconstruction and introduce how to train our convolutional neural networks together used in the proposed method. To the best of our knowledge, this is the first work using convolutional neural network for multiple description coding.

A. Framework

Our multiple description coding framework has three components: MDGN network, standard codec of JPEG, MDRN network, as depicted in the Fig. 1. The MDGN network \( g(I, \omega) \) is responsible to generate diverse descriptions \( I_A \) and \( I_B \) from the ground-truth image \( I \) with size of \( M \times N \). Here, \( \omega \) is the parameter set of MDGN network and other networks’ parameter set can be defined in this similar way. Due to the widely usage of standard codec, such as JPEG, the standard-compatible coding framework becomes significant for practical applications. Thus, we use the JPEG codec to compress these descriptions so that image redundancy can be further reduced to get the lossy descriptions \( \hat{I}_A \) and \( \hat{I}_B \). The JPEG compressions of \( I_A \) and \( I_B \) are respectively represented as \( \hat{I}_A = c(I_A, \phi) \), and \( \hat{I}_B = c(I_B, \phi) \), where \( c(\cdot) \) is the compression function of codec. The compressed description streams are separately transmitted over different channels. However, image compression with standard codec often incurs coding artifacts. Thus, MDRN network, denoted as reconstruction function \( R(\cdot) \), is leveraged to remove these artifacts for image enhancement and enlarge the lossy description so that the final reconstruction image is guaranteed to have the same size with the ground-truth image \( I \). Finally, the receiver can still decode the received packet to get a description for acceptable quality reconstruction \( \hat{I}_A \) with SRNA network or \( \hat{I}_B \) with SRNB network, even though any one description is missing, as displayed in the Fig. 1. If both descriptions are received, high quality reconstruction \( \hat{I} \) can be built by CRN network.

Fig. 1. The framework of multiple description coding based on deep convolutional neural networks
As we all know, it’s not easy to jointly train the MDGN network and MDRN network, because the quantization function in the codec of lossy compression is non-differentiable. Thus, the reconstruction error from the MDRN network can’t be directly back-propagated to the MDGN network. Following our previous work [33], we learn the MDVCN network to imitate the two consecutive procedures of codec’s compression and description’s reconstruction with MDRN network. As a result, we can train our whole framework in an end-to-end fashion.

B. Objective function

The objective function for our multiple description coding framework is written as follows:

$$
\arg\min_{\omega, \alpha, \theta} L_{MDGN}(I_A, I_B, I, \omega) + L_{MDRN}(I_A, I_B, I, \alpha) + L_{MDVCN}(I_A, I_B, \theta),
$$

where three losses for training are respectively the loss of MDGN network, the loss of MDRN network, and MDVCN network’s loss.

$$
L_{MDGN}(I_A, I_B, I, \omega) = L_{SSIM}(u(I_A), I) + L_{SSIM}(u(I_B), I) + \beta L_{dis}(I_A, I_B),
$$

where three losses for training are respectively the loss of MDGN network, the loss of MDRN network, and MDVCN network’s loss.

The loss of $L_{MDGN}(I_A, I_B, I, \omega)$ is used to supervise the learning of the parameters $\omega$ of the MDGN network in Eq. (3), where $u(\cdot)$ is the linear up-sampling function and $\beta$ balances the contributions between descriptor’s SSIM loss [34] and distance loss, which are in effect contradictory to a certain extent. In addition, $QF$ is the quality factor for JPEG compression and $\text{clip}(\cdot)$ is the clip function to restrict value between $\kappa 1$ and $\kappa 2$ (e.g., $\kappa 1 = 5 \times 10^{-3}$ and $\kappa 2 = 5 \times 10^{-2}$). Hence, the parameter of $\beta$ plays a significant role on generating valid multiple descriptions. Note that the better quality is encoded, when the larger $QF$ is set for JPEG.

On one hand, we hope that the two produced descriptions structurally similar to the input image so that the decoded descriptions can be watched directly for receiver, even without the processing of MDRN network. Consequently, SSIM loss function is used to supervise each description’s learning. For example, the SSIM for description $I_A$ is defined as follows:

$$
L_{SSIM}(u(I_A), I) = \frac{1}{M \cdot N} \sum_i L_{SSIM}(u(I_A)i, I_i),
$$

where $\mu_{u(I_A)}$ and $\sigma_{u(I_A)}^2$ respectively denote the mean value and the variance of the neighborhood window centered by pixel $i$ in the image $u(I_A)$. Similarly, $\mu_I$, as well as $\sigma_I^2$, is denoted in this way. $\sigma_{u(I_A)i}$, $\sigma_{u(I_A)}$, and $\sigma_{u(I_A)i}$ are the covariance between neighbourhood windows centered by pixel $i$ in the image $u(I_A)$ and in the image $I$. Meanwhile, $c_1$ and $c_2$ are two constant values (e.g., $c_1 = 1 \times 10^{-4}$, and $c_2 = 9 \times 10^{-4}$). As a matter of fact, the calculation of mean value is a special kind of convolution, which is also named by average pooling, while variance operation actually involve twice operations of average pooling. It’s obvious that the function of SSIM in Eq. (4-5) is differentiable, so the SSIM error can be efficiently back-propagated via optimization.

On the other hand, according to the Gamal and Cover theorem of [35][36], the MDGN network should pledge to have mutual information between two generated descriptions so that we can receive a acceptable reconstruction, even when only one description is got at the client. It’s obvious that SSIM loss function keeps the two descriptions yielded by the MDGN network structurally similar. In the meantime, the two produced descriptions by neural networks are used as opposing labels to regularize the training of MDRN network. Consequently, the high-quality central reconstruction with two diverse descriptions can be guaranteed. Contrary to the SSIM loss, the distance loss function is utilized to keep the detail difference between two descriptions, which is written as:

$$
L_{dis}(f(I_A), f(I_B)) = - \frac{1}{M \cdot N} \sum_i (||I_{A_i} - I_{B_i}||_L).
$$

For brevity latter, the content loss function and gradient difference loss function between two images $X$ and $Y$ are defined as:

$$
L_c(X, Y) = \frac{1}{M \cdot N} \sum_i (||X_i - Y_i||_L),
$$

$$
L_{gd}(X, Y) = \frac{1}{M \cdot N} \sum_i (||\nabla_s X_i - \nabla_s Y_i||_L),
$$

where $\nabla_s$ is the $s$-th gradient between each pixel and $s$-th pixels among 8-neighbourhood $\Omega$. Here, L1-norm is chosen to produce sharper results than L2-norm, which has been reported in [37][38].

In the MDRN network, both content loss $L_c$ and gradient difference loss $L_{gd}$ supervise the learning of side reconstruction $R(c(I_A, \phi), \alpha_1)$, $R(c(I_B, \phi), \alpha_2)$ and central reconstruction $R(c(I_A, \phi), c(I_B, \phi), \alpha_3)$, which is presented as follows:

$$
L_{MDRN}(I_A, I_B, I, \alpha) = L_c(I, R(c(I_A, \phi), \alpha_1)) + L_{gd}(I, R(c(I_A, \phi), \alpha_1))
+ L_c(I, R(c(I_B, \phi), \alpha_2)) + L_{gd}(I, R(c(I_B, \phi), \alpha_2))
+ L_c(I, R(c(I_A, \phi), c(I_B, \phi), \alpha_3)) + L_{gd}(I, R(c(I_A, \phi), c(I_B, \phi), \alpha_3)).
$$

In order to back-propagate the error from the MDRN network to the MDGN network, we learn MDVCN network to approximate the procedure from the lossless descriptions to the lossy description reconstruction. Both content loss and
gradient difference loss are used to regularize the training of MDVCN network, which are given as follows:

\[ L_{MDVCN}(I_A, I_B, \theta) = L_c(\tilde{I}_A, V(I_A, \theta_1)) + L_{gd}(\tilde{I}_A, V(I_A, \theta_1)) + L_c(\tilde{I}_B, V(I_B, \theta_2)) + L_{gd}(\tilde{I}_B, V(I_B, \theta_2)) + L_c(\tilde{I}, V(I_A, I_B, \theta_3)) + L_{gd}(\tilde{I}, V(I_A, I_B, \theta_3)). \]  

(10)

In addition to the aforementioned loss, we use MDVCN network to explicitly supervise the learning of the MDGN network or directly use gradient from MDVCN network as the gradient approximation from the standard codec. It’s worth noticing that MDVCN network does not use any more, once the whole training is finished, that is to say, only the MDRN network and the MDGN network during the testing are respectively leveraged to create multiple descriptions for compression and reconstruct these descriptions.

### C. Network architecture

The MDGN network is composed with eight convolutional layers, which has one input stream, but two output streams, that is to say, the extracted feature maps with feature extraction network (FEN) from layer 1-4 are shared by generator network-A (GNA) and generator network-B (GNB). The FEN network has four convolutional layers, whose first layer’s spatial kernel size is \(9 \times 9\) and other layers’ is \(3 \times 3\). In the GNA and GNB networks, there are four convolutional layers with spatial kernel size \(3 \times 3\) except for the last layer with \(9 \times 9\). The large spatial kernel \(9 \times 9\) of convolutional layer in the first layer and last layer could further enlarge the receptive field of convolutional networks on the basis of small kernel \(3 \times 3\). Hence, image’s context information is well considered during the generation of descriptions. The details about each layer in the MDGN network are listed in the Table I, where “\(k\)” represents the kernel size, “c-in” denotes the number of channel input, “c-out” is the total output map’s number in the corresponding layer. Meanwhile, “conv” represents convolutional layer and “deconv” indicates the de-convolutional layer. From this table, it can be seen that all the layers employ stride step 1 except for the second convolutional layer with stride of 2. All the convolutional layers are activated by the ReLU activation function apart from the last layer in the MDGN network.

The MDRN network consists of SRNA network, SRNB network, and CRN network. In fact, we can let SRNA network and SRNB network share the same parameter set. Meanwhile, CRN network uses the outputs from the SRNA-network and SRNB-network to reconstruct the central images. But, in order to better back-propagate the errors
from the MDRN network to the previous networks and avoid too deep networks for central reconstruction, we use three separate networks without cross connection and no weights sharing to respectively reconstruct side images and central image. They all use the eight convolutional layers. Seven convolutional layers and one deconvolution layer are used in the MDRN network so as to remove the coding artifacts and up-scale feature maps to the full-resolution at the same time. The obvious difference between them is that CRN network has two lossy descriptions as input while the two other networks only have one lossy descriptions as input. All the details are specified in Table II, from which we can observe that the first and last convolutional layers use the $9 \times 9$ spatial kernel to ensure the receptive field large enough, so that more spatial features are captured to better reconstruct the degraded descriptions. In addition, all the convolutional layers are activated by the ReLU, but the last layers of SRNA network, SRNB network, and CRN network are processed without any activation.

As described above, MDVCN network bridges the gap between MDGN network and MDRN network so that the errors of the reconstruction can be properly back-propagated from MDRN network to MDGN network. MDVCN network and MDRN network are designed to have same structure, because they can be seen as the same class of low-level image processing problems by learning. Thus, we have three virtual networks for MDVCN network: virtual side reconstruction network-A (VSRNA), virtual side reconstruction network-B (VSRNB), and virtual central reconstruction network (VCRN), whose network structures in the Table III are similar to the one’s of MDRN network in the Table II. However, the inputs of MDVCN network and MDRN network are different, in which the former one takes the decoded lossy descriptions $\hat{I}_A$ and $\hat{I}_B$ as inputs, while the latter one is fed with lossless multiple descriptions $I_A$ and $I_B$.

D. Network learning

Obviously, it’s challenging to learn our whole framework directly, but our problem of learning multiple description neural networks can be separated into several sub-problems learning. In order to resolve these problems, we provide two learning ways for error back-propagation. These two ways are presented in the following and respectively referred to as learning algorithm-1 and learning algorithm-2. Our learning algorithm-1 treats MDVCN network as feature function to build the reconstruction by fixing the parameter of MDVCN network so that reconstruction errors from MDVCN network can be back-propagated for the supervision of the MDGN network ahead of standard codec. It means that the MDGN network and the MDRN network are trained separately. On the contrary, our learning algorithm-2 uses MDVCN network’s back-propagated error for MDGN network to approximately estimate the error from the codec without fixing any network’s parameter, when explicitly training the MDGN network and the MDRN network simultaneously. The details about these two learning algorithms will be described next.

\textbf{Algorithm 1 Learning Multiple Description Neural Networks}

\textbf{Input:} Ground truth image: $I$; the number of iteration: $R$; the total number of images for training: $n$; the batch size during training: $m$.

\textbf{Output:} The parameter sets of MDGN network and MDRN network: $\omega$, $\alpha$;

1. Initialize to produce multiple descriptions $I_A$ and $I_B$ by down-sampling for preparation of the training of MDRN network;
2. Initialize parameter sets: $\omega$, $\alpha$, $\phi$, $\theta$;
3. for $r = 1$ to $R$
   4. Compute multiple descriptions $I_A$ and $I_B$ by standard codec with $\phi$;
   5. for $epoch = 1$ to $p$
      6. for $i = 1$ to floor($n/m$)
         7. Update the parameter set of $\alpha$ to train the MDRN network according to the minimization of the Eq. (9) with $i$-th batch images;
      8. end for
   9. end for
10. Generate the multiple descriptions reconstruction dataset $\hat{I}_A$ and $\hat{I}_B$ with the parameter set of $\alpha$ by the MDRN network;
13. for $epoch = 1$ to $p$
14. for $j = 1$ to floor($n/m$)
15. Update the parameter set of $\theta$ by training MDVCN network to minimize the Eq. (10) with $j$-th batch images from $I_A$ and $I_B$ dataset;
16. end for
17. end for
18. Generate the multiple descriptions images $I_A$ and $I_B$ with the parameter set of $\omega$ by the MDGN network;
24. for $epoch = 1$ to $q$
25. for $i = 1$ to floor($n/m$)
26. Update the parameter set of $\alpha$ by training the MDRN network to minimize the Eq. (9) with $i$-th batch images;
27. end for
28. end for
30. Update the parameter set of $\alpha$ by training the MDRN network to minimize the Eq. (9) with $i$-th batch images;
31. end for
32. end for
33. end for
34. return $\omega$, $\alpha$;

1) Learning algorithm-1: To back-propagate the error from the MDRN network to MDGN network, we decompose the learning problem of MDGN network, MDRN network and MDVCN network once in Eq. (1) into three separate subproblem learning, but they depends on each other closely. Specifically, we first initialize all the parameter sets mentioned previously, and multiple descriptions $I_A$ and $I_B$ dataset by down-sampling for the training of MDRN network and compress this dataset. Secondly, the parameter set of $\alpha$ is updated by training MDRN network based on minimization of the Eq. (9). Then, we generate multiple descriptions reconstruction images $\hat{I}_A$, $\hat{I}_B$, and $\hat{I}$ dataset with the parameter set of $\alpha$ of MDRN network. This reconstruction dataset $\hat{I}_A$, $\hat{I}_B$, and $\hat{I}$ can be used to train MDVCN network by updating the parameter set of $\theta$ based on the minimization of the Eq. (10). Next, we update the parameter set of $\omega$ with fixed $\theta$ to train MDGN network according to the minimization of the Eq. (3) and Eq. (9). After training MDGN network, the multiple descriptions images $I_A$ and $I_B$ are generated with the parameter set $\omega$ of MDGN network and then start the next iteration. The details about learning algorithm-1 are summarized in the Algorithm-1.

2) Learning algorithm-2: Different from our learning algorithm-1, we separate the whole framework learning as two sub-problem learning: the sub-problem of simultaneously learning MDGN network and MDRN network, and the learning sub-problem of MDVCN network. Concretely, the parameter sets of MDGN network and
MDRN network: $\omega$, $\alpha$ are learned by the optimization with gradient descent method at the same time. After feeding input data into MDGN network to produce multiple descriptions $I_A$ and $I_B$ and compressing them with standard codec, MDRN network are used to reconstruct these compressed multiple descriptions $\hat{I}_A$ and $\hat{I}_B$. Meanwhile, the lossless multiple descriptions $I_A$ and $I_B$ are fed into MDVCN network. This is feed-forward propagation of our deep convolution neural networks, but the error from the MDRN network is blocked by the codec. Here, we can explicitly use the error from MDVCN network as the approximate error from the codec. Thus, we can simultaneously update MDGN network and MDRN network. The whole process is detailed in the Algorithm-2.

**Algorithm 2 Learning Multiple Description Neural Networks**

**Input:** Ground truth image: $I$; the number of iteration: $T$; the total number of images for training: $n$; the batch size during training: $m$; 

**Output:** The parameter sets of MDGN network and MDRN network: $\omega$, $\alpha$; 

1: Initialize parameter sets: $\omega$, $\alpha$, $\phi$, $\beta$; 
2: Pre-train MDVCN network; 
3: for $i = 1$ to $T$ do 
4: for epoch $= 1$ to $l$ do 
5: for $i = 1$ to floor($n/m$) do 
6: $a$) : Generate multiple descriptions $I_A$ and $I_B$ with the the parameter set of $\omega$; Then, compress multiple descriptions $I_A$ and $I_B$ with standard codec with $\phi$; 
7: $b$) : Update the parameter set of $\omega$ and $\alpha$ by training the MDGN network and MDRN network simultaneously by minimizing Eq. (3) and Eq. (9) with $i$-th batch images. Note that the MDGN network uses errors from $MDVCN$ network for back-propagation to update parameter set of $\omega$; 
8: $c$) : Generate multiple descriptions reconstruction images $\hat{I}_A$ and $\hat{I}_B$ with the parameter set of $\alpha$ by MDRN network; 
9: $d$) : Update the parameter set of $\alpha$ by training MDVCN network based on minimization of Eq. (10) with $i$-th batch images with $\hat{I}_A$ and $\hat{I}_B$; 
10: end for 
11: end for 
12: return $\omega$, $\alpha$;

After comparing learning algorithm-1 with learning algorithm-2, we can see that the training stability of the second one relies on whether pre-trained MDVCN network is well trained or not. Meanwhile, this network also has great impacts on the learning of MDGN network, because the bad accuracy of approximated error propagation from MDVCN network will results in the insufficiency of multiple description generation. On the contrary, the first algorithm is more easily implemented in any neural network platform, because there is no any changes in the process of neural network’s optimization. Meanwhile, the performance of learning algorithm-1 tends to be more stable than the second one due to the reliable dependency among three neural networks. It comes from a fact that the good training of MDRN network will directly lead to the good training of MDVCN network, and then MDVCN network will give a supervision of the MDGN network. Conclusively, both of them can resolve the learning problem of multiple description neural networks, but the learning algorithm-1 is more practical, so we use it to illustrate the efficiency of the whole framework in the experimental sections.

**III. EXPERIMENTAL RESULTS**

We evaluate the proposed method against eight baselines with state-of-the-art artifacts removal techniques [22,25] and advanced super-resolution based on very deep convolutional neural networks, such as [31,32]. Note that there are 20 convolutional layers used for super-resolution in [31,32]. Four baselines “MDB1a-MDB4a” are formed with the techniques of artifacts removal [22,25] and very deep convolutional neural network based super-resolution [31]. Meanwhile, super-resolution of [22] are combined with artifacts removal [22,25] to build four other baselines “MDB1b-MDB4b”. Furthermore, in order to fully demonstrate the efficiency of the proposed method, we form a baseline model, which is denoted as “Our-base”, when replacing MDGN network to generate multiple descriptions with the poly-phase down-sampling technique in [19]. For simplicity, the proposed method is marked as “Ours”. Besides, the training of the proposed framework will be in detail described next.

**A. Training data and implementation details**

Our whole framework is implemented in the platform of TensorFlow [39] with Algorithm-1. The 400 images with size 180x180 from [40] are used as our training data-set, which are augmented by cropping, flipping, and rotating image to build our training data set. There are the total number 3200 of image patches with size of 160x160 used for our framework’s training. Four images in Fig. 2 are used to evaluate the efficiency of the proposed method for testing. Our framework is trained with the Adam optimization method [41]. The parameters for Adam optimization are set to be $\beta_1 = 0.9$, $\beta_2 = 0.999$. The learning rate of training is initially set as 0.0001, but the learning rate decays to be half of the initial one when the training step reaches 3/5 of total step. Once the training step reaches 4/5 of total step, it reduces to be 1/4 of the initial one. The multiple descriptions are compressed by standard JPEG codec with $QF$ to be 2, 6, 10, 20, and 40 for the proposed framework during the training and testing. The multiple descriptions for MDB1a-MDB4a and MDB1b-MDB4b as well as “Our-base” are compressed with the $QF$ set 2, 3, 4, 10, and 50.

**B. Comparisons with several baselines**

To validate the efficiency of the proposed framework, we employ the Peak Signal to Noise Ratio (PSNR) and SSIM to measure the objective quality. The multiple description artifacts removal results with Fois [14], BM3D [22], DiCTV [17] and CONCOLOR [18] are got with strict usage of the authors open codes according to the parameter settings in their papers. Meanwhile, for image super-resolution in
Fig. 3. The side reconstruction and central reconstruction objective measurement comparison on PSNR and SSIM for several state-of-the-art approaches. (a1,b1) are respectively the side-reconstruction PSNR results of image (a) and (b) in Fig. 2, (a2,b2) are the central reconstruction PSNR results of image (a) and (b) in Fig. 2, (a3,b3) are respectively the side reconstruction SSIM results of image (a) and (b) in Fig. 2, (a4,b4) are the central reconstruction SSIM results of image (a) and (b) in Fig. 2.

[31, 32], we use their official provided model to enlarge these multiple description after artifacts removal so as to guarantee the advances of eight baselines, when comparing with the proposed method. From the comparison in Fig. 3 and Fig. 4, it can be seen that ours-base has better performance on SSIM for the side reconstruction and central reconstruction against eight baselines MDB1a-MDB4a and MDB1b-MDB4b in the full range. In the
most cases, the PSNR measurement of ours-base is better than eight baselines MDB1a-MDB4a and MDB1b-MDB4b. Only at the very low bit-rate, the PSNR of ours-base has slight smaller than MDB4a and MDB4b, but our-base with higher SSIM measurement has priority than MDB4a and MDB4b.

This comes from that the structural preservation of image is more significant than detail preservation at the very low bit-rate, as shown in Fig. 3 and Fig. 4.

Compared to the ours-base, the proposed method has more PSNR and SSIM gains in the most cases, especially at the
Fig. 5. The visual comparison of different methods for (a) in Fig. 2. (a1) input image (left) and the enlargement in the red line boxed region of input image (right), (a2) multiple description created by a poly-phase down-sampling technique, (a3) multiple description generated by the proposed MDGN network, (a4) left image is the difference between a pair of image in (a2) and the right image is difference between a pair of image in (a3), (a5) is the compressed image in (a2), (a6) is the compressed image in (a3); (b-f) description reconstruction images, where the (b1-b3, b4-f4, b5-f5) are the side reconstruction images, the (b3-f3) and (b6-f6) are the central reconstruction images; (b1-b3) MDB1a(24.574/0.714/0.292(s) and 27.849/0.778/0.583(c)), (b4-b6) MDB1b(24.637/0.715/0.292(s) and 27.849/0.778/0.583(c)), (c1-c3) MDB2a(25.202/0.725/0.292(s) and 28.088/0.769/0.583(c)), (c4-c6) MDB2b(25.318/0.727/0.292(s) and 28.215/0.771/0.583(c)), (d1-d3) MDB3a(25.399/0.723/0.292(s) and 28.129/0.763/0.583(c)), (d4-d6) MDB3b(25.507/0.726/0.292(s) and 28.231/0.765/0.583(c)); (e1-e3) MDB4a(25.785/0.748/0.292(s) and 25.969/0.754/0.583(c)), (e4-e6) MDB4b(25.847/0.749/0.292(s) and 26.051/0.756/0.583(c)); (f1-f3) Ours-base(28.641/0.787/0.292(s) and 30.870/0.831/0.583(c)), (f4-f6) Ours(28.577/0.803/0.292(s) and 31.410/0.842/0.585(c)). (Note that the red line boxed regions in (b-f) represent the part regions enlarged from the corresponding full resolution images like (a1); the real image size of (a2-a6) is half of input image’s size, while all the other images have the same size as the input image).

Although the literature of [32] has reported that their approach has greater PSNR gains than [31] for general image super-resolution, the performance of [32] is slight better than the one’s of [31], when these super-resolution approaches are used for description’s resolution enhancement after artifacts removal, which can be found in Fig. 3 and Fig. 4 when comparing MDB1a-MDB4a with MDB1b-MDB4b.

We have compared the visual quality of the proposed method with different methods’ for multiple description coding based on deep convolutional neural networks, which is displayed in Fig. 5 and Fig. 6. In these figures, MDB1a(24.574/0.714/0.292(s) and 27.849/0.778/0.583(c)) represents the measurements of PSNR/SSIM/bpp for side reconstruction and central reconstruction based on the approach of MDB1a. Similarly, other methods can be denoted in this way. Our MDGN network-produced descriptions, as displayed in Fig. 5(a3) and Fig. 6(a3), maintain more important details than the ones generated with the poly-phase down-sampling technique [19], even after image compression. The differences between these pairs of descriptions are exhibited in Fig. 5(a4) and Fig. 6(a4), from which it can be observed that the proposed method tends to have high bit-rate. Because the proposed method in this paper focuses on the appearance similarity but details difference for multiple descriptions generation without apparent structural distance loss to regularize the training at the very low bit-rate, the proposed method has a little lower PSNR gains than ours-base in some cases. For the improvement of the proposed method at low bit-rate, the first way is to replace direct description distance loss with structural distance loss during training. Another feasible way is to employ 4x resolution reduction when generating the descriptions with MDGN network and compressing descriptions at the very low bit-rate, but larger QF is used for the proposed method like our previous work [33].
Fig. 6. The visual comparison of different methods for (d) in Fig. 2. (a1) input image (left) and the enlargement in the red line boxed region of input image (right), (a2) multiple description created by a poly-phase down-sampling technique, (a3) multiple description generated by the proposed MDGN network, (a4) left image is the difference between a pair of image in (a2) and the right image is difference between a pair of image in (a3), (a5) is the compressed image in (a2), (a6) is the compressed image in (a3); (b-f) description reconstruction images, where the (b1-f1, b2-f2) and (b4-f4, b5-f5) are the side reconstruction images, the (b3-f3) and (b6-f6) are the central reconstruction images; (b1-b3) MDB1a(27.808/0.824/0.232(s) and 31.319/0.861/0.463(c)), (b4-b6) MDB1b (27.843/0.825/0.232(s) and 31.367/0.862/0.463(c)), (c1-c3) MDB2a(28.524/0.832/0.232(s) and 31.526/0.857/0.463(c)), (c4-c6) MDB2b(28.579/0.833/0.232(s) and 31.601/0.858/0.463(c)), (d1-d3) MDB3a(28.642/0.827/0.232(s) and 31.288/0.850/0.463(c)), (d4-d6) MDB3b(28.700/0.828/0.232(s) and 31.352/0.851/0.463(c)); (e1-e3) MDB4a(29.244/0.846/0.232(s) and 29.428/0.850/0.463(c)), (e4-e6) MDB4b(29.270/0.846/0.232(s) and 29.471/0.850/0.463(c)); (f1-f3) Ours-base(32.040/0.865/0.232(s) and 33.098/0.881/0.463(c)), (f4-f6) Ours(31.913/0.874/0.229(s) and 33.865/0.889/0.458(c)). (Note that the red line boxed regions in (b-f) represent the part regions enlarged from the corresponding full resolution images like (a1); the real image size of (a2-a6) is half of input image’s size, while all the other images have the same size as the input image)

keep the description distance on the details and has less structural difference preservation. Furthermore, the descriptions from our MDGN network tend to highlight obvious feature pixels for all the descriptions in order to protect the key features. Therefore, the protected feature of lossy descriptions always can be kept, although they are possibly badly smoothed and contaminated by compression, as shown in Fig. 5(a5-a6) and Fig. 6(a5-a6).

The side reconstruction images and central reconstruction images have been displayed in Fig. 5(b-f) and Fig. 6(b-f). From these figures, it can be clearly seen that the side reconstruction images and central reconstruction images with the proposed method look more natural and have more detail preservation than eight baselines MDB1a-MDB4a, MDB1b-MDB4b, and our-base. Our-base has better performance than the eight baselines. Among these baselines, both MDB4a and MDB4b keep more details than MDB1a-MDB3a, MDB1b-MDB3b, which can be seen in Fig. 5(b-e) and Fig. 6(b-e). From the above objective and visual comparisons, it can be concluded that it’s very important to emphasize on significant context features when automatically generating appearance-similar but details-different descriptions with convolutional neural networks, as compared to the poly-phase down-sampling technique. Meanwhile, the better descriptions always benefit the better side and central description reconstruction.

IV. CONCLUSION

In this paper, we introduce multiple description image coding based on deep convolutional neural networks. First, multiple description network is employed to automatically yield valid multiple descriptions. Then, these multiple descriptions are compressed by standard codec so that our whole framework is compatible with standard codec. Thirdly, we use multiple description reconstruction network to enhance these descriptions and restore them to be full resolution for the reconstruction of the compressed multiple descriptions. Besides, two learning algorithms are provided to train our whole framework. Moreover, both distance loss and SSIM loss are combined together to train the multiple description generator networks in order to make sure that the generated multiple descriptions are diverse, but they have shared structures information.
REFERENCES

[1] Y. Wang, A. Reibman, and S. Lin, “Multiple description coding for video delivery,” Proceedings of the IEEE, vol. 93, no. 1, pp. 57–70, 2005.
[2] V. Goyal, “Multiple description coding: compression meets the network,” IEEE Signal Processing Magazine, vol. 18, no. 5, pp. 74–93, 2001.
[3] K. Viswanatha, E. Akyol, and K. Rose, “Combinatorial message sharing and a new achievable region for multiple descriptions,” IEEE Transactions on Information Theory, vol. 62, no. 2, pp. 769–792, 2016.
[4] H. Bai, W. Lin, M. Zhang, A. Wang, and Y. Zhao, “Multiple description video coding based on human visual system characteristics,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 24, no. 8, pp. 1390–1394, 2014.
[5] M. Liu and C. Zhu, “Enhancing two-stage multiple description scalar quantization,” IEEE Signal Processing Letters, vol. 16, no. 4, pp. 253–256, 2009.
[6] V. Vaishampayan, “Design of multiple description scalar quantizers,” IEEE Transactions on Information Theory, vol. 39, no. 3, pp. 821–834, 1993.
[7] S. S., K. Ramchandran, V. Vaishampayan, and K. Nahrstedt, “Multiple description wavelet based image coding,” IEEE Transactions on Image Processing, vol. 9, no. 5, pp. 813–826, 2000.
[8] Y. Xu and C. Zhu, “End-to-end rate-distortion optimized description generation for H. 264 multiple description video coding,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 23, no. 9, pp. 1523–1536, 2013.
[9] C. Tian and J. Chen, “New coding schemes for the symmetric k-description problem,” IEEE Transactions on Information Theory, vol. 56, no. 10, pp. 5344–5365, 2010.
[10] V. Vaishampayan, N. Sloane, and S. Servetto, “Multiple-description vector quantization with lattice codebooks: Design and analysis,” IEEE Transactions on Information Theory, vol. 47, no. 5, pp. 1718–1734, 2001.
[11] V. Goyal, J. Kelner, and J. Kovacevic, “Multiple description vector quantization with a coarse lattice,” IEEE Transactions on Information Theory, vol. 48, no. 3, pp. 781–788, 2002.
[12] H. Bai, C. Zhu, and Y. Zhao, “Optimized multiple description lattice vector quantization for wavelet image coding,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 17, no. 7, pp. 912–917, 2007.
[13] Y. Wang, M. Orchard, V. Vaishampayan, and A. Reibman, “Multiple description coding using pairwise correlating transforms,” IEEE Transactions on Image Processing, vol. 10, no. 3, pp. 351–366, 2001.
[14] I. Bajic and J. Woods, “Concatenated multiple description coding of frame-rate scalable video,” in International Conference on Image Processing, New York, 2002.
[15] Y. Zhang, M. Motani, and H. Garg, “Wireless video transmission using multiple description codes combined with prioritized DCT compression,” in International Conference on Multimedia and Expo, Lausanne, Aug. 2002.
[16] J. Apostolopoulos, “Error-resilient video compression through the use of multiple states,” in IEEE International Symposium on Image Processing, Vancouver, Sep. 2000.
[17] C. Kim and S. Lee, “Multiple description coding of motion fields for robust video transmission,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 11, no. 9, pp. 999–1010, 2001.
[18] S. Cho and W. Pearlman, “A full-featured, error-resilient, scalable wavelet video codec based on the set partitioning in hierarchical trees (SPIHT) algorithm,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 12, no. 3, pp. 157–171, 2002.
[19] N. Franchi, M. Fumagalli, R. Lancini, and S. Tubaro, “Multiple description video coding for scalable and robust transmission over IP,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 15, no. 3, pp. 321–334, 2005.
[20] T. Tillo, M. Grangetto, and G. Olmo, “Multiple description image coding based on Lagrangian rate allocation,” IEEE Transactions on Image Processing, vol. 16, no. 3, pp. 673–683, 2007.
[21] T. Tillo and G. Olmo, “A novel multiple description coding scheme compatible with the JPEG2000 decoder,” IEEE Signal Processing Letters, vol. 11, no. 11, pp. 908–911, 2004.
[22] A. Foi, V. Katkovnik, and K. Egiazarian, “Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images,” IEEE Transactions on Image Processing, vol. 16, no. 5, pp. 1395–1411, 2007.
[23] H. Chang, M. Ng, and T. Zeng, “Reducing artifacts in JPEG decompression via a learned dictionary,” IEEE Transactions on Signal Processing, vol. 62, no. 3, pp. 718–728, 2014.
[24] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, “Image denoising by sparse 3-D transform-domain collaborative filtering,” IEEE Transactions on Image Processing, vol. 16, no. 8, pp. 2080–2095, 2007.
[25] J. Zhang, R. Xiong, C. Zhao, Y. Zhang, S. Ma, and W. Gao, “CONCOLOR: Constrained non-convex low-rank model for image deblocking,” IEEE Transactions on Image Processing, vol. 25, no. 3, pp. 1246–1259, 2016.
[26] L. Zhao, H. Bai, A. Wang, Y. Zhao, and B. Zeng, “Two-stage filtering of compressed depth images with markov random field,” Signal Processing: Image Communication, vol. 51, pp. 11–22, 2017.
[27] C. Dong, Y. Deng, L. Change, and X. Tang, “Compression artifacts removal by a deep convolutional network,” in IEEE Conference on Computer Vision and Pattern Recognition, Boston, Jun. 2015.
[28] L. Galteri, L. Seidenari, M. Bertini, and A. DelBimbo, “Deep generative adversarial compression artifact removal,” in arXiv:1704.02518, 2017.
[29] L. Cavigelli, P. Hager, and L. Benini, “CAS-CNN: A deep convolutional neural network for image
compression artifact suppression,” in *IEEE International Joint Conference on Neural Networks*, Anchorage, May 2017.

[30] L. Zhao, H. Bai, A. Wang, and Y. Zhao, “Iterative range-domain weighted filter for structural preserving image smoothing and de-noising,” *Multimedia Tools and Applications*, pp. 1–28, 2017.

[31] J. Kim, L. Kwon, and L. Mu, “Accurate image super-resolution using very deep convolutional networks,” in *IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, Jun. 2016.

[32] K. Zhang, Y. Chen, Y. Chen, D. Meng, and L. Zhang, “Beyond a gaussian denoiser: residual learning of deep cnn for image denoising,” *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142 – 3155, 2017.

[33] L. Zhao, H. Bai, A. Wang, and Y. Zhao, “Learning a virtual codec based on deep convolutional neural network to compress image,” in *arXiv:1712.05969*, 2017.

[34] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[35] A. Gamal and T. Cover, “Achievable rates for multiple descriptions,” *IEEE Transactions on Information Theory*, vol. 28, no. 6, pp. 851–857, 1982.

[36] L. Lastras and V. Castelli, “Near sufficiency of random coding for two descriptions,” *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 681–695, 2006.

[37] M. Mathieu, C. Couprie, and Y. LeCun, “Deep multi-scale video prediction beyond mean square error,” in *arXiv: 1511.05440*, 2015.

[38] L. Zhao, J. Liang, H. Bai, A. Wang, and Y. Zhao, “Simultaneously color-depth super-resolution with conditional generative adversarial network,” in *arXiv: 1708.09105*, 2017.

[39] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, and et al., “Tensorflow: large-scale machine learning on heterogeneous distributed systems,” in *arXiv:1603.04467*, 2016.

[40] Y. Chen and T. Pock, “Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1256–1272, 2017.

[41] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *arXiv:1412.6980*, 2014.