Predictive and Adaptive Deep Coding for Wireless Image Transmission in Semantic Communication

Wenyu Zhang, Haijun Zhang, *Fellow, IEEE*, Hui Ma, Hua Shao, Ning Wang, *Member, IEEE*, and Victor C. M. Leung, *Life Fellow, IEEE*

***Abstract*— Semantic communication is a newly emerged communication paradigm that exploits deep learning (DL) models to realize communication processes like source coding and channel coding. Recent advances have demonstrated that DL-based joint source-channel coding (DeepJSCC) can achieve exciting data compression and noise-resiliency performances for wireless image transmission tasks, especially in environments with low channel signal-to-noises (SNRs). However, existing DeepJSCC-based semantic communication frameworks still cannot achieve adaptive code rates for different channel SNRs and image contents, which reduces its flexibility and bandwidth efficiency. In this paper, we propose a predictive and adaptive deep coding (PADC) framework for realizing flexible code rate optimization with a given target transmission quality requirement. PADC is realized by a variable code length enabled DeepJSCC (DeepJSCC-V) model for realizing flexible code length adjustment, an Oracle Network (OraNet) model for predicting peak-signal-to-noise (PSNR) value for an image transmission task according to its contents, channel signal to noise ratio (SNR) and the compression ratio (CR) value, and a CR optimizer aims at finding the minimal data-level or instance-level CR with a PSNR quality constraint. By using the above three modules, PADC can transmit the image data with minimal CR, which greatly increases bandwidth efficiency. Simulation results demonstrate that the proposed DeepJSCC-V model can achieve similar PSNR performances compared with the state-of-the-art Attention-based DeepJSCC (ADJSCC) model, and the proposed OraNet model is able to predict high-quality PSNR values with an average error lower than 0.5dB. Results also demonstrate that the proposed PADC can use nearly minimal bandwidth consumption for wireless image transmission tasks with different channel SNR and image contents, at the same time guaranteeing the PSNR constraint for each image data.**

***Index Terms*— Adaptive code length, joint source channel coding, quality prediction, semantic communication, wireless image transmission.**

Manuscript received 18 July 2022; revised 28 October 2022; accepted 1 January 2023. Date of publication 11 January 2023; date of current version 14 August 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62102021, Grant 62101030, Grant 62225103, and Grant U22B2003; in part by the China Post-Doctoral Science Foundation under Grant 2020M680350; in part by the Beijing Natural Science Foundation under Grant L212004; in part by the China University Industry-University-Research Collaborative Innovation Fund under Grant 2021FNA05001; in part by the Guangdong Pearl River Talent Recruitment Program under Grant 2019ZT08X603; in part by the Guangdong Pearl River Talent Plan under Grant 2019JC01X235; in part by the Shenzhen Science and Technology Innovation Commission under Grant R2020A045; and in part by the Fundamental Research Funds for the Central Universities of USTB under Grant FRF-IDRY-20-020. The associate editor coordinating the review of this article and approving it for publication was C. Huang. *(Corresponding author: Haijun Zhang.)*

Wenyu Zhang, Hui Ma, and Hua Shao are with the School of Intelligence Science and Technology, Institute of Artificial Intelligence, University of Science and Technology Beijing, Beijing 100083, China (e-mail: wyzhang@ustb.edu.cn; hui_ma@ustb.edu.cn; shaohua@ustb.edu.cn).

Haijun Zhang is with the Beijing Advanced Innovation Center for Materials Genome Engineering and the Beijing Engineering and Technology Research Center for Convergence Networks and Ubiquitous Services, University of Science and Technology Beijing, Beijing 100083, China (e-mail: haijunzhang@ieee.org).

Ning Wang is with the Henan Joint International Research Laboratory of Intelligent Networking and Data Analysis, School of Information Engineering, Zhengzhou University, Henan 450001, China (e-mail: ienwang@zzu.edu.cn).

Victor C. M. Leung is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China, and also with the Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, BC V6T 1Z4, Canada (e-mail: vleung@ieee.org).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TWC.2023.3234408.

Digital Object Identifier 10.1109/TWC.2023.3234408

I. Introduction

In a wireless communication system, source coding and channel coding are two commonly used techniques for improving communication efficiency and reliability respectively. In general, source coding is usually conducted at the application layer, and it is helpful for decreasing the data amount by reducing the information redundancy of the original source data. For image data compression tasks, traditional source coding methods mainly use the following three steps to reduce data amount: transform, quantization and entropy coding, such as the well-known Discrete Cosine Transform (DCT) based Joint Photographic Experts Group (JPEG) [1] and Better Portable Graphics (BPG) [2] standards, and Discrete Wavelet Transform (DWT) based JPEG2000 standard [3]. Channel coding generally works at the physical layer, and it is mainly used for improving the transmission reliability by using error check or error correcting coding techniques, such as Low Density Parity Check Code (LDPC) [4], Polar code [5], and Turbo code [6]. In conventional communication systems, the above two coding processes are implemented as two separate functional modules or blocks, and this practice is helpful for designing, developing, and maintaining the communication system in a flexible way.

From the perspective of system-level optimality, the above separable or layered source-channel coding paradigm is actually suboptimal because it is designed and optimized independently without considering the mutual impacts, thus cannot work cooperatively to achieve the best overall communication capacity [7]. For example, the source decoding block assume that its input code from channel decoding block is error-free, and then it can recover the transmitted data without error from the result of channel decoding. However, when channel SNR is low and channel decoding is not able to guarantee

the zero bit error rate, the quality of the recovering result will be dramatically reduced. This problem is called as the 'cliff-effect', and it widely exists in conventional wireless communication systems [8], [9], [10].

To solve the above problem, joint source-channel coding (JSCC) has been proposed by jointly designing and optimizing source coding and channel coding process to achieve system-level optimality for the coding process [11], [12], [13]. Due to the advances of deep learning (DL) in recent years, it is practical for us to use the powerful data compression and noise resiliency capabilities of DL models to realize the end-to-end JSCC process with one DL model [14], [15], [16], and recent advances have demonstrated that DL based JSCC (DeepJSCC) methods, particularly deep autoencoder based JSCC methods, can achieve exciting capacities in both data compression and transmission reliability for various wireless data transmission tasks, such as image [9], text [17], and speech [18].

For wireless image transmission tasks, in the conventional communication paradigm, an image may be firstly compressed by using the JPEG/JPEG2000 source coding methods and then transformed as noise-resiliency channel code by using a channel coding method like LDPC. At the receiver, the transmitted bits can be recovered without errors by using the corresponding channel decoding and source decoding processes when channel SNR is not too low. In the framework of DeepJSCC, the source channel coding and decoding processes are jointly completed by using an encoder and a decoder respectively, and the encoder-decoder pair (EDP) is trained in an end-to-end way, along with a non-trainable physical channel. In this way, the whole wireless transmission process is optimized from the original input image data and recovered output image data, thus it is globally optimal, and recent works have shown that this DeepJSCC-enabled wireless transmission scheme can achieve superior data compression performance compared with traditional JPEG/JPEG2000 methods. In addition, it can achieve significant reliability performance improvement against channel noise compared with conventional communication scheme, especially in low SNR regime [9], [19], [20].

In [9], deep convolutional neural network based autoencoders are firstly used to realize joint source-channel coding for wireless image transmission, and recent advances have shown that the capability of DeepJSCC models can be further enhanced by using channel output feedback [10] and refinement transmission methods [21], [22]. However, as a new emerged methodology, the above proposed DeepJSCC models suffer from the following two problems:

• **SNR-adaptation problem**: Since each DeepJSCC model is trained with a specific channel SNR, the obtained DeepJSCC models are channel-dedicated, and they can achieve the best performance for wireless image transmission tasks only when the channel SNRs are similar to their corresponding training SNRs. For example, a DeepJSCC model trained with channel SNR equals to 12dB only performs well when the test SNR is close to 12dB, otherwise, the performance will be greatly decreased. Since the channel SNR can vary in both low and high SNR regimes, we need to prepare several DeepJSCC models to cover the full SNR regime.

• **CR-adaptation problem**: The capacity of a DeepJSCC model is also influenced by the CR of the obtained semantic codes. Similar to the SNR-adaptation problem, a DeepJSCC model is trained with one dedicated CR, thus the length of the obtained semantic code (SC) is fixed, and cannot be changed as other code lengths. However, in practice, it is necessary for us to dynamically adjust the code length to save bandwidth for different channel SNR and image contents. To meet this requirement, we need to prepare several DeepJSCC models with different CRs for adjusting the CRs.

Due to the above two problems, one needs to train multiple DeepJSCC models to cover all channel SNR and CR regimes. For example, if we divide the channel SNR as the following 8 levels $\{0, 3, 6, 9, 12, 15, 18, 21\}$dB, and divide the CR as the following 5 levels $\{0.1, 0.2, 0.3, 0.4, 0.5\}$, we need to prepare $5 \times 8 = 40$ DeepJSCC models, which greatly increases the training and deployment cost of DeepJSCC models. Moreover, even if we prepare enough DeepJSCC models trained with different channel SNRs and CRs, we still cannot realize adaptive rate control for a specific image transmission task with a quality constraint because the quality of the recovered image is unknown to us.

To solve the SNR-adaptation problem, Xu et al. [23] proposed an attention module-based DeepJSCC model (ADJSCC) that can adjust the learned image features in different channel SNR conditions. ADJSCC is motivated by the compression ratio and channel coding rate optimization method in [24], and utilizes a squeeze-and-excitation module [25] to adjust the learned image features in different SNR conditions to maximize the image reconstruction quality. More specifically, except for the commonly used convolutional blocks (Conv-blocks) and trans-convolutional blocks (TransConv-blocks), an attention feature block (AFB) is introduced to adaptively adjust the signal strength of the learned features after a Conv-block or TransConv-block, such that the encoder and decoder can adjust the coding resources assigned to source coding/decoding process and channel coding/decoding process. When channel SNR is high, the coding resources assigned to channel coding/decoding can be relatively lower, and more resources are assigned for source coding/decoding to improve data compression efficiency. On the contrary, when channel SNR is low, more coding resources can be used for channel coding/decoding to enhance transmission reliability. Accordingly, the coding resources assigned to channel coding are reduced. Simulation results demonstrate that the achieved PSNR performance of one single ADJSCC model is the same as the highest PNSRs achieved by multiple JSCC models with different training SNRs. Therefore, ADJSCC can effectively solve the SNR-adaptation problem.

To solve the CR-adaptation problem, Yang et al. [26] proposed a DeepJSCC method with adaptive rate control capability (DeepJSCC-A) for wireless image transmission. In DeepJSCC-A, the source coding and channel coding processes are conducted by using two separate encoders, and the output semantic codes are divided into the following two parts: selective features that can be either active or nonactive for transmission, and non-selective features that are always active for transmission. A policy network is introduced

for determining which selective features are transmitted along with the non-selective features. Simulation results demonstrated that the DeepJSCC-A can achieve adaptive rates according to different channel SNR and image contents. However, DeepJSCC-A still suffered from the following disadvantages: First, DeepJSCC-A cannot support arbitrary CRs when selecting the features, and the rate choices are quite limited. Secondly, the non-selective features will always occupy a large portion of the coding resources, thus the coding rates cannot be lower than a threshold, e.g. 0.25 in the paper [26], which further reduces the rate choices. Thirdly, though DeepJSCC-A can adjust the rate according to the channel SNR and image contents, it is not able to predict the PSNR qualities of the transmitted images, thus the transmission quality cannot be flexibly controlled with a target PSNR constraint. Fourthly, the simulation results on CIFAR10 data show that DeepJSCC-A cannot achieve equivalent performance compared with ADJSCC trained with specific CRs, especially in high channel SNR regime. As a conclusion, we can see that the CR-adaptation problem is still not effectively addressed by DeepJSCC-A.

In conventional communication systems, it is a common practice that the transmitter adjusts the transmission data rate according to the real-time channel conditions. For example, when using LDPC as the channel code, we can use a relatively lower data rate to enhance the noise-resilience capacity when channel SNR is relatively low [27]. In addition, using a smaller modulation order is also helpful for improving transmission reliability. Realizing the above adaptive rate control process relies on the knowledge of how the transmission performance, such as bit error rate (BER) and block error rate (BLER), is influenced by the channel SNR, and this knowledge can be previously formulated or known to us. However, in semantic communication, adaptive rate control still cannot be well addressed even if we have DeepJSCC models with flexible rate adjustment abilities, because the transmission quality is jointly influenced by the channel SNR, CR, and data contents, and the relationship between the transmission quality and the three factors are very complex and hard to be formulated. If we want to optimize the transmission rate for each single image transmission task, we must know the transmission qualities by jointly considering the influences of the three factors, otherwise, the transmission quality cannot be guaranteed. Therefore, Except for the SNR-adaptation and CR-adaptation problems, we also need to solve the transmission quality prediction problem to enable the transmitter to realize transmission quality-guaranteed rate control for every single transmission.

To address the above problems, in this paper, we propose a predictive and adaptive deep coding (PADC) framework for achieving flexible, adaptive, and quality guaranteed rate control according to different channel SNR and image contents. Compared with ADJSCC [23] and DeepJSCC-A [26], the proposed PADC framework has following contributions:

- Unlike the code rate selection framework of DeepJSCC-A, in PADC we propose a simple, flexible, and effective variable code length enabled DeepJSCC model, namely DeepJSCC-V, that can achieve arbitrarily code rates for different channel SNRs and image contents. More specifically, by using a simple code mask operation and with a given compression ratio (CR) R, DeepJSCC-V simply selects the first R-proportion semantic features as the transmitted codes, and the remained features will be directly abandoned. In this way, the CR-adaptation operation is very simple in implementation, and the training process is similar to ADJSCC that uses randomly generated CR values for different training samples, but doesn't need additional training tricks.
- To solve the transmission quality prediction problem, we propose an Oracle Network (OraNet) that is able to predict high-quality PSNR for a single image transmission task by jointly considering channel SNR, CR, and image contents. By utilizing this capability, OraNet can be further used for predicting the optimal CR value with given channel SNRs and image contents, and the optimized CR can be used to guide the DeepJSCC-V model to adjust the code length.
- With the help of DeepJSCC-V and OraNet models, we propose data-level CR optimization and instance-level optimization in the framework of PADC. Data-level CR optimization problem aims at minimizing the CR of all image data with a constraint of ensuring that the average PSNR of all images is not lower than a given target threshold value. Instance-level optimization is a new problem and has not been investigated in existing work, it aims at minimizing the CR of every single image, at the same time guarantees its transmission quality. Both the two optimization processes can be used for improving the bandwidth efficiency of DeepJSCC-enabled wireless image transmissions.
- We use the CIFAR10 data and ImageNet data for training the proposed DeepJSCC-V and OraNet models, and test the performances of the proposed PADC framework by using CIFAR100 and Kodak datasets. Results obtained demonstrate that the proposed DeepJSCC-V model can achieve similar PSNR performances and higher transmission efficiency compared with the state-of-the-art ADJSCC model [23], and results also show that the proposed OraNet can obtain high-quality PSNR prediction results according to the image contents, channel SNR, and CR. Particularly, by querying the OraNet, results obtained demonstrate that the proposed PADC find nearly minimal CR configuration that meets the PSNR requirements.

The remainder of the paper is organized as follows: and Section 2 illustrates the framework of the proposed PADC framework, Section 3 presents the simulation results of the proposed method, along with the network architectures of the used DeepJSCC-V and OraNet models. Finally, Section 4 concludes this paper.

Notation: $\mathbb{C}^n$ and $\mathbb{R}^n$ means the complex and real data with total dimensions n respectively. $x \sim \mathcal{CN}(\mu_x, \sigma_x^2)$ designates variable x follows a circularly-symmetric complex Gaussian distribution with mean μ_x and σ_x^2. Variables with bold fonts means that they are matrices or vectors. $\mathbf{a} \cdot \mathbf{b}$ means the dot-product of vectors $\mathbf{a}$ and $\mathbf{b}$. $\mathbf{x}^*$ means the conjunct

transpose of complex variable $\mathbf{x}$. At last, $\mathbb{E}[x]$ means the expectation value of variable x, and $\mathbf{I}$ denotes the identity matrix.

II. Predictive and Adaptive Deep Coding

In this section, we elaborate the framework proposed PADC method, including the system model, PADC framework, model training, and adaptive rate control.

A. System Model

We consider a DeepJSCC-enabled wireless image transmission system, and our goal is to reconstruct the original target image data with minimal distortion at the receiver from the received SCs under certain channel SNR and data compression ratio conditions. The DeepJSCC model is composed of a trainable encoder E_ϕ, and non-trainable physical channel, and a trainable decoder D_θ, where ϕ and θ denote the Encoder parameters and Decoder parameters respectively. Denote the N-dimensional original data that needs to be transmitted as $\mathbf{x} \in \mathbb{R}^N$, it will be firstly encoded as a K-dimensional complex semantic code (SC) $\mathbf{z} \in \mathbb{C}^K$, as given by

$$\mathbf{z} = E_\phi(\mathbf{x}, \gamma, R), \tag{1}$$

where γ means the channel SNR, and $R = K/N$ denotes the CR of the input image data. After power normalization, the SC obtained will be transmitted over the physical channel and subsequently received by the receiver, and corrupted with additive white Gaussian noise (AWGN), as given by

$$\tilde{\mathbf{z}} = h\mathbf{z} + \mathbf{n}, \tag{2}$$

where $h \in \mathcal{C}$ denotes the channel gain coefficient, and $\mathbf{n} \sim \mathcal{CN}(0, \sigma^2\mathbf{I})$ denotes independent identically distributed (i.i.d) AWGN samples with power σ^2. For the AWGN channel without considering the influence of channel gain h, the transmitting model is simply given by

$$\tilde{\mathbf{z}} = \mathbf{z} + \mathbf{n}. \tag{3}$$

At last, the decoder recovers data from the received SC $\tilde{\mathbf{z}}$, and the result can be formulated as

$$\tilde{\mathbf{x}} = D_\theta(\tilde{\mathbf{z}}, \gamma, R). \tag{4}$$

In the above DeepJSCC-enabled semantic communication process, the channel SNR has a significant influence on the quality of the final reconstructed image $\tilde{\mathbf{x}}$. In general, the channel SNR is determined by the noise power σ^2, which is mainly caused by the nature noise and interferences in the physical channel. In this paper, we suppose that physical channel noise is AWGN, and to quantify the quality of the physical channel, we define the SNR as follows

$$\gamma = 10\log_{10}\frac{P_z}{\sigma^2}, \tag{5}$$

where P_z denotes the average power of the SCs, and in practice, we will use a power normalization operation to ensure that it is equal to a bounded value, e.g. $P_z = 1$.

To meet this power constraint, the SC $\mathbf{z}$ needs to be normalized to satisfy the condition $\frac{1}{K}\mathbb{E}[\mathbf{z}^*\mathbf{z}] \le P_z$, where $\mathbf{z}^*$ denotes the conjunct transpose of $\mathbf{z}$. In general, a physical channel with lower SNR means relatively higher physical noise power compared with the SC power, the quality of the final recovered result will be lower compared with a relatively higher physical channel SNR.

B. The Proposed PADC Framework

To deal with the CR-adaptation and SNR-adaptation problems at the same time, as Figure 1 shows, in this paper we propose a novel DeepJSCC framework called predictive and adaptive deep coding (PADC), and it is composed of a variable code length enabled DeepJSCC (DeepJSCC-V) model for joint source-channel coding with flexible code length adjustment ability, an Oracle Network (OraNet) for predicting the quality of the reconstructed image with specific channel SNR and compression ratio (CR) information, and a CR optimizer that minimizes the transmission data rate with given PSNR constraint. The design principles of the above three modules are illustrated as follows:

1) DeepJSCC-V: The design principle of the network architecture of DeepJSCC-V is the same as ADJSCC, which utilizes an attention feature module for adjusting the learned features according to different channel SNRs, such that the Encoder and Decoder models can solve the SNR adaptation problem. The main difference is that the proposed DeepJSCC-V model has an SC mask process for solving the CR-adaptation problem, and it trains the model to adjust the deep coding process with different CR in an automatic way. We denote the SC obtained from the Encoder as $\mathbf{z}^o = E_\phi(\mathbf{x}, \gamma)$, and $\mathbf{z}^o \in \mathbb{C}^{K_{\max}}$, where $K_{\max}$ denotes the maximal allowable SC length. $K_{\max}$ can be set as a relatively large value such that we can have more CR choices, and in this paper, we suggest setting it as $K_{\max} = N/2$, in this way the maximal CR value is 0.5. Then, SC $\mathbf{z}^o$ is masked by using a binary SC mask vector (SCMV), denoted as $\boldsymbol{\alpha} \in \{0,1\}^{K_{\max}}$, to adjust its length. More specifically, given an expected CR $R \in (0, 1]$, we can know the corresponding code length is $K = \lceil RN/2 \rceil$, where $\lceil \cdot \rceil$ means round-up operation. Note that a complex semantic symbol is represented by two real-value semantic features obtained from the Encoder. Then, the value of element α_i is set as follows

$$\alpha_i = \begin{cases} 1, & \text{if } i \le K \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

Then, the SC mask operation is given by

$$\mathbf{z} = \mathbf{z}^o \cdot \boldsymbol{\alpha}, \tag{7}$$

where $\cdot$ denotes the dot product operation. When $\alpha_i = 1$, the i-th code or symbol will be selected for transmission. In other words, the first K elements of SC will be selected as masked SC (MSC), and then transmitted to the receiver. When the MSC is received by the receiver, it needs to be expanded with 0 elements to ensure that its dimension is still $K_{\max}$. We can simply pad $K_{\max} - K$ zero elements behind the received MSC, and then the expanded SC can be sent to the decoder to reconstruct the input image. We denote the received SC as $\tilde{\mathbf{z}}$, then we can have the following padded SC $\tilde{\mathbf{z}}^o = [\tilde{\mathbf{z}}, \mathbf{0}]^T \in \mathbb{C}^{K_{\max}}$. Finally, the recovered result can be

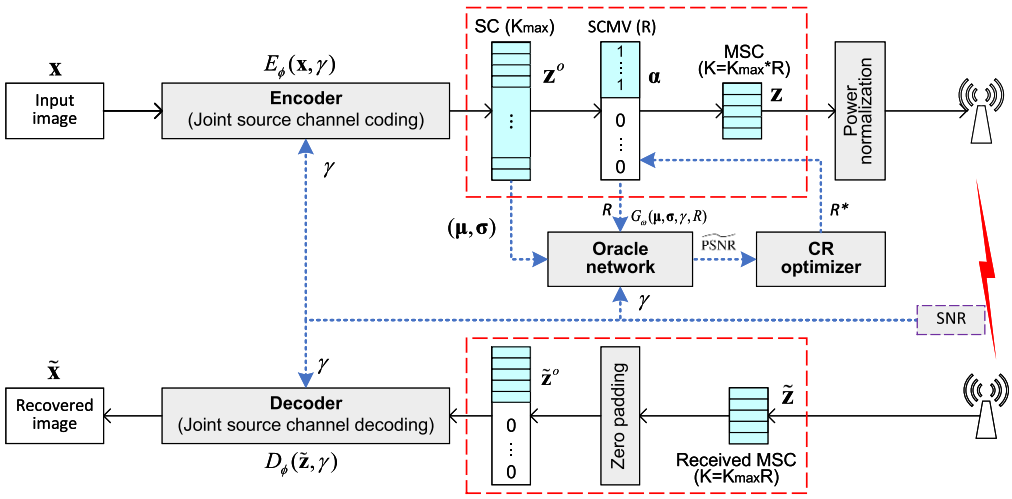


Fig. 1. The proposed PADC framework.

obtained as $\tilde{\mathbf{x}} = D_\theta(\tilde{\mathbf{z}}^o, \gamma)$. Note that, in DeepJSCC-V, the dimensions of the Encoder output $\mathbf{z}^o$ and Decoder input $\tilde{\mathbf{z}}^o$ are all fixed as $K_{\max}$, thus CR R is not included as a variable of the Encoder and Decoder.

The output $\mathbf{z}^o$ of the encoder also can be a 3-dimensional data with shape $N_H \times N_W \times N_c$, where N_H, N_W and N_c denote the width, height, and channel respectively. In this situation, the SCMV $\boldsymbol{\alpha}$ can be a binary vector with length N_c, and given a code length K, the first K channels will be selected for transmission. For example, when $K = 8$, which means that the first 8 channels will be selected as the MSC, and the remained channels will be abandoned.

Fig. 2 shows the architecture of the used JSCC-V network, which is composed of the following three blocks: residual convolutional block (RCB), attention feature block (AFB), and residual transpose convolutional block (RTCB). The residual structure helps enhance the joint source-channel coding capability, at the same time improving the convergence speed of the model training process [28]. The normalization operation used after a convolutional or transconvolutional layer is local divisive normalization based generalized normalization transformation (GDN), which is effective for dealing with image compression and density modeling tasks [29]. A convolutional or transconvolutional layer is designated by parameters $m \times n \times c | \updownarrow s$, in which m, n, and c correspond to the width, height, and output channel of the used convolutional kernel, symbols $\downarrow$ and $\uparrow$ means the downsampling operation and up-sampling operation respectively, and parameter s means the stride length. The architecture of AFB is the same as the AFBs in [23] and [25].

After the last AFB of the Encoder, we can obtain semantic feature data with shape $H/4 \times W/4 \times 48$, and then we can know the maximal code length is $K_{\max} = 3HW/2$. Note that the SCs are complex symbols. Given a specified CR R, we can know the first $2K = \lceil 2\ RK_{\max} \rceil$ channels will be selected as the MSC. The MSC will be flattened as a vector and transformed as a K-dimensional complex SC, i.e. $\mathbf{z} \in \mathbb{C}^K$. In the same way with [9], the complex SC is normalized by the following equation $\widehat{\mathbf{z}} = \sqrt{KP_z}\frac{\mathbf{z}}{\mathbf{z}^*\mathbf{z}}$, where $\mathbf{z}^*$ denotes the conjunct transpose of $\mathbf{z}$, and P_z denotes the average transmitting power constraint. In this way, the average transmitting power of the SC can satisfy the following constraint $\frac{1}{K}\mathbb{E}[\widehat{\mathbf{z}}^*\widehat{\mathbf{z}}] \le P_z$.

2) OraNet: The OraNet is a network that aims at predicting the quality of a reconstructed image by jointly considering channel SNR γ, CR R, and SC $\mathbf{z}^o$. Particularly, the SC $\mathbf{z}^o$ is the extracted feature of the original input image, and note that its total dimension can be different for input images with different sizes. Compared with the original input image data, SC $\mathbf{z}^o$ is simpler compression of the image content for the OraNet to learn the relationship between the quality of the reconstructed images and the image content. If we use the original image data for predicting the quality value, we will need a much bigger model to understand the complex image contents, and the computation complexity will be greatly increased. To further reduce complexity and keep the dimension stable, in this paper we will use channel-wise mean (CWM) $\boldsymbol{\mu}_z$ and channel-wise standard deviation (CWSTD) $\boldsymbol{\sigma}_z$ of SC $\mathbf{z}^o$ as the image content features for predicting the quality of the reconstructed image. For example, suppose that the shape of SC $\mathbf{z}^o$ is $48 \times 8 \times 8$, i.e. it has 48 channels, and in each channel, the width and height are all 8. Accordingly, we can know the dimension of $\boldsymbol{\mu}_z$ and $\boldsymbol{\sigma}_z$ are all 48, and in total, we have 96 image content feature values. In the Simulation section, we will further provide some examples to demonstrate why CWM and CWSTD features can be used for predicting image qualities.

The image quality metric can be set as any suitable metric that can quantify the quality of the final reconstructed images. In this paper, we use the commonly used peak signal-to-noise ratio (PSNR) as the image quality metric, and it formulates the distortion between the input image $\mathbf{x}$ and reconstructed image $\tilde{\mathbf{x}}$ as follows:

$$\text{PSNR} = 10\log_{10}\frac{\text{MAX}^2}{\text{MSE}}, \tag{8}$$

where MAX means the allowable peak pixel value of the images. Since the images used is 8-bit per pixel, we can know the signal peak value is $\text{MAX} = 2^8 - 1 = 255$. A higher PSNR means better quality of the recovered image because the MSE becomes lower. Denote G_ω as an OraNet model parametrized

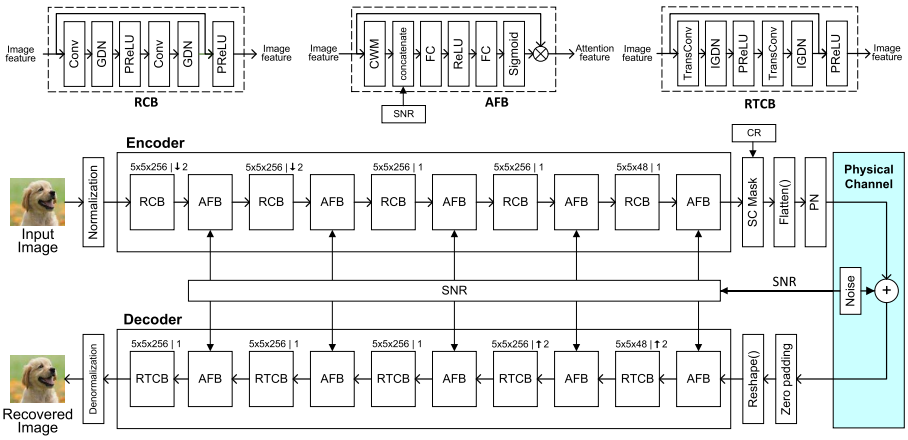


Fig. 2. Architecture of the used DeepJSCC-V model, in which PN means power normalization.

by ω, then the predicted image quality can be represented as $\widetilde{\text{PSNR}} = G_\omega(\boldsymbol{\mu}_z, \boldsymbol{\sigma}_z, \gamma, R)$.

Figure 3 shows the architecture of the used OraNet model, which is mainly composed of two residual fully connection blocks (RFCBs). With the two RFCBs, the model can effectively extract the representative information for predicting the PSNR, and a fully connection layer can be used for obtaining the predicted PSNR outputs. A ReLU activation function is added behind the FC layer because we want to ensure that the predicted PSNR values are larger than 0. In general, the complexity of OraNet is much lower than the DeepJSCC-V model, thus its inference efficiency is much higher.

3) CR Optimizer: With the trained DeepJSCC-V and OraNet models, we can minimize CR values to improve the bandwidth efficiency with a given PSNR constraint. Due to the difference of PSNR constraint, the CR optimization problem can be divided into the following two levels:

a) Data-level CR optimization: Suppose that the number of SNR levels and CR levels are N_{SNR} and N_{CR} respectively, we can obtain a $N_{SNR} \times N_{CR}$ sized PNSR performance table $Q_D(\gamma, R)$ by using the test data samples. In this problem, the PSNR constraint is ensuring that the expected PSNR value of all reconstructed images is lower than a target PSNR value PSNR_{th}, as given by

$$\begin{aligned} \mathbf{P}1. \quad & \min \ R \\ & \text{s.t. } Q_D(\gamma, R) \geq \text{PSNR}_{th}, \\ & \quad 0 < R \leq 1. \end{aligned} \tag{9}$$

Since we only have $N_{SNR} \times N_{CR}$ feasible solutions, the above problem can be readily solved by an exhaustive search process. In this way, we can obtain an optimized CR value for all transmitted images regardless of the influences of the image contents. The above problem is quite similar to the CR-adaptation problem in conventional transmission systems that optimizes the coding rates according to wireless channel conditions, and the image content is not needed in this scenario. With one trained DeepJSCC-V model, we can realize the data-level CR optimization for DeepJSCC-enabled wireless image transmission tasks. We can also realize data-level optimization if we have enough ADJSCC models trained with different CRs, but it will cause much higher training and deployment costs compared with DeepJSCC-V.

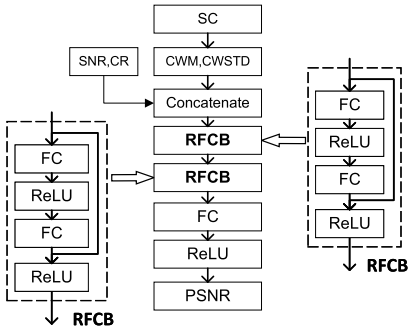


Fig. 3. Architecture of the used OraNet model.

b) Instance-level CR optimization: In this problem, the PSNR constraint is guaranteeing that the PSNR of every single reconstructed image is not lower than a target PSNR value PSNR_{th}. We can use an OraNet model for predicting the PSNR values by jointly considering CWM feature $\boldsymbol{\mu}_z$, CWSTD feature $\boldsymbol{\sigma}_z$, channel SNR γ, and CR R, then the optimization model is given by

$$\begin{aligned} \mathbf{P}2. \quad & \min \ R \\ & \text{s.t. } G_\omega(\boldsymbol{\mu}_z, \boldsymbol{\sigma}_z, \gamma, R) \geq \text{PSNR}_{th}, \\ & \quad 0 < R \leq 1. \end{aligned} \tag{10}$$

Since the CWM $\boldsymbol{\mu}_z$, CWSTD $\boldsymbol{\sigma}_z$, and channel SNR γ are all given, and $G_\omega(\boldsymbol{\mu}_z, \boldsymbol{\sigma}_z, \gamma, R)$ is a monotonically increasing function to R, i.e. when data compression ratio R is not sufficiently large, a relative larger value of R will produce a relative higher PSNR value. As such, the above problem also can be easily solved by querying the OraNet. More specifically, we can divide the CR regime as the following L CR levels $R_1, \ldots, R_L$, then we can compute the corresponding PSNR predictions for all the CR levels, and choose the minimal CR R^* that satisfies the PSNR constraint as the optimization result. It is also practical to use a bisection search process to find a more accurate CR value. With the optimized CR R^*, the DeepJSCC-V model then can use a minimal data rate to transmit the image data with PSNR constraint.

It is necessary to emphasize that, instance-level CR optimization is a new problem and has not been considered before.

In conventional sparable communication systems, it is not applicable for us to adjust CR by considering the influences of the image contents, but it can be realized by the proposed PADC-enabled semantic communication systems. Particularly, since instance-level CR optimization enables us to flexibly adjust the CR for every single image, thus it can achieve higher bandwidth efficiency compared with data-level CR optimization, at the same time guaranteeing the PSNR constraint of every transmission. However, it doesn't mean instance-level CR optimization can be a better choice than data-level CR optimization in all situations. When it is not required for us to guarantee the transmission quality of each single image, it is also practical for us to use data-level CR optimization to improve bandwidth efficiency, and only using DeepJSCC-V can achieve this goal.

C. Model Training

In PADC, there are two models need to be trained, i.e. the DeepJSCC-V model and the OraNet model, and their training processes are illustrated as follows:

1) DeepJSCC-V: Given an N-dimensional input image data $\mathbf{x}$, we can obtain the SC as $\mathbf{z}^o = E_\phi(\mathbf{x}, \gamma)$, and then obtain the MSC $\mathbf{z} \in \mathbb{C}^K$ by using the SCMV with a given CR R. After transmission, the receiver obtains SC $\tilde{\mathbf{z}}$, and then we get the final reconstructed image as $\tilde{\mathbf{x}} = D_\theta(\tilde{\mathbf{z}}, \gamma, R)$. The loss function is defined as the mean square error (MSE) between the original image $\mathbf{x}$ and the reconstructed image $\tilde{\mathbf{x}}$, as given by

$$\mathcal{L}_{\theta,\phi}(\mathbf{x}, \tilde{\mathbf{x}}; \gamma, R) = \frac{1}{N}\|\mathbf{x} - \tilde{\mathbf{x}}\|^2. \tag{11}$$

With the above loss function, the model parameters ϕ, θ can be optimized by minimizing the MSE-based distortion degree between the input images and output images, as given by

$$(\phi^*, \theta^*) = \text{argmin}_{\phi,\theta}\ \mathbb{E}_{\mathbf{x},\gamma,R}\left[\mathcal{L}_{\theta,\phi}(\mathbf{x}, \tilde{\mathbf{x}}; \gamma, R)\right]. \tag{12}$$

In the training process, the channel SNR γ and CR R of each image sample are randomly generated. In this paper, the channel SNR and CR are randomly generated from intervals $\gamma \in [0, 27]$dB and $R \in [0.05, 0.5]$ respectively. With sufficient training samples, the DeepJSCC-V model can automatically learn the optimal source-channel coding process with specific SNR γ and CR R.

The process of one training round for DeepJSCC-V is illustrated in Algorithm 1. Note that the whole training process may contain thousands of training rounds, and it will not be terminated until the loss reaches convergence. In each training round, we randomly generated different SNR and CR values, and by using the AFB proposed in [23] and the SC mask process proposed in this paper, we can obtain a DeepJSCC-V model trained with SNR-adaptation and CR-adaptation capabilities. Particularly, by using the SC mask process, an SC symbol with an inner location is more probable to be transmitted, and it will carry more information compared with other outsider SC symbols. For example, suppose that SC $\mathbf{z}^o$ is a 3-dimensional tensor, for two SC channels $\mathbf{z}_i^o, \mathbf{z}_j^o \in \mathbf{z}^o$, if $i > j$, then the i-th channel data $\mathbf{z}_i^o$ will be more probable to be selected for transmission compared with the j-th channel data $\mathbf{z}_j^o$. During the training process, the DeepJSCC-V model will automatically learn to compress more information on $\mathbf{z}_i^o$ to reduce the loss objective as much as possible. As a result, an inner SC channel will be trained to carry more information compared with other outsider channels.

Algorithm 1 One Training Round for DeepJSCC-V

Input: Training dataset $\mathcal{X}$, batch size B, learning rate η;
Output: DeepJSCC-V model parameters (ϕ, θ);
Sample a data batch $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_B]$ from $\mathcal{X}$;
for each data sample $\mathbf{x}_i \in \mathbf{X}$:
  Randomly generate channel SNR $\gamma_i \in \mathcal{U}(0, 27)$;
  Randomly generate CR $R_i \in \mathcal{U}(0.05, 0.5)$;
  Compute SC $\mathbf{z}_i^o = E_\phi(\mathbf{x}_i, \gamma_i)$;
  Generate SCMV $\boldsymbol{\alpha}_i$ by Eq. (6);
  Compute MSC $\mathbf{z}_i$ by Eq. (7);
  if channel is fading:
    Randomly generate CSI $h_i \sim \mathcal{CN}(0, 1)$;
    Randomly generate AWGN $\mathbf{n}_i \sim \mathcal{CN}(0, \sigma_i^2\mathbf{I})$ with SNR γ_i;
    Compute the received MSC $\tilde{\mathbf{z}}_i$ by Eq. (2);
  else if channel is AWGN:
    Randomly generate AWGN $\mathbf{n}_i \sim \mathcal{CN}(0, \sigma_i^2\mathbf{I})$ with SNR γ_i;
    Compute the received MSC $\tilde{\mathbf{z}}_i$ by Eq. (3);
  end if
  Recover image by $\tilde{\mathbf{x}}_i = D_\theta(\tilde{\mathbf{z}}_i, \gamma_i, R_i)$;
end for each
Compute average loss $\mathcal{L} = \frac{1}{NB}\sum_{i=1}^{B}\|\mathbf{x}_i - \tilde{\mathbf{x}}_i\|^2$;
Update model parameters (ϕ, θ) by gradient descent.

2) OraNet: Once the DeepJSCC model is trained, we can then use it for training the OraNet model. With the obtained SC $\mathbf{z}^o$, we can then compute its CWM $\boldsymbol{\mu}_z$ and CWSTD $\boldsymbol{\sigma}_z$ features. Accordingly, we can also obtain the reconstructed image $\tilde{\mathbf{x}}$, and compute the PSNR between $\tilde{\mathbf{x}}$ and $\mathbf{x}$ by using equation (8). Except for the ground-truth PSNR value, we can also obtain a predicted $\widetilde{\text{PSNR}}$ value by using the OraNet, i.e. $\widetilde{\text{PSNR}} = G_\omega(\boldsymbol{\mu}_z, \boldsymbol{\sigma}_z, \gamma, R)$. In this way, the loss function of OraNet can be defined as

$$\mathcal{L}_\omega(\boldsymbol{\mu}_z, \boldsymbol{\sigma}_z, \gamma, R) = (\text{PSNR} - \widetilde{\text{PSNR}})^2. \tag{13}$$

In the same way with the DeepJSCC model, the model parameters ω can be optimized by minimizing the above loss function, as given by

$$\omega^* = \text{argmin}_\omega\ \mathbb{E}_{\mathbf{z}^o,\gamma,R}\left[\mathcal{L}_\omega(\boldsymbol{\mu}_z, \boldsymbol{\sigma}_z, \gamma, R)\right]. \tag{14}$$

In each training iteration, the channel SNR γ and CR R are randomly generated in the same way as the training process of the DeepJSCC-V model. Algorithm 2 illustrates one training round for the OraNet. Note that, DeepJSCC-V and OraNet are trained independently. More specifically, the OraNet model is trained on the basis of a previously trained DeepJSCC-V model because it needs to use the SC and PSNR as the training data. In this way, we can know that an OraNet is dedicated to a specific DeepJSCC-V model, and it is not suggested to use it for predicting the transmission qualities of other

DeepJSCC-V models that are trained with different channel types and training data.

Algorithm 2 One Training Round for OraNet

Input: Training dataset $\mathcal{X}$, batch size B, learning rate η;
Output: OraNet model parameters ω;
Sample a data batch $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_B]$ from $\mathcal{X}$;
for each data sample $\mathbf{x}_i \in \mathbf{X}$:
  Randomly generate channel SNR $\gamma_i \in \mathcal{U}(0, 27)$;
  Randomly generate CR $R_i \in \mathcal{U}(0.05, 0.5)$;
  Call DeepJSCC-V to obtain $\mathbf{z}_i^o$ and $\hat{\mathbf{x}}_i$;
  Compute CWM $\boldsymbol{\mu}_z^{(i)}$ and CWSTD $(\boldsymbol{\sigma}_z^{(i)})^2$ of $\mathbf{z}_i^o$;
  Compute $\widetilde{\text{PSNR}}_i = G_\omega(\boldsymbol{\mu}_z^{(i)}, \boldsymbol{\sigma}_z^{(i)}, \gamma_i, R_i)$;
  Compute real PSNR_i with $\mathbf{x}_i$ and $\hat{\mathbf{x}}_i$ by Eq. (8);
end for each
Compute average loss $\mathcal{L} = \frac{1}{B}\sum_{i=1}^{B}(\text{PSNR}_i - \widetilde{\text{PSNR}}_i)^2$;
Update model parameters ω by gradient descent.

III. Simulation Results

In this section, we will first illustrate the simulation settings, then present the performance evaluation results to demonstrate that the proposed PADC framework can achieve predictive and adaptive rate control for wireless image transmission, and some examples will also be provided for understanding the properties of the proposed PADC.

A. The Used Datasets and Simulation Settings

We will first use CIFAR10 and CIFAR100 data [30] as the experimental data for training and testing the performance of the proposed method. CIFAR10 data and CIFAR100 data are two similar datasets, but the classes of the two datasets are 10 and 100 respectively. Since we only focus on transmitting the images, thus the class labels will not be used. Both of the two datasets contain 60000 color image samples with size 32×32, of which 50000 of them are training images, and the remaining 10000 images are test samples. In this paper, we will use the CIFAR10 training samples and test samples as the training dataset and validation dataset respectively, and they are used in the training process for obtaining the DeepJSCC models. When evaluating the performance, we will use CIFAR100 test samples as the test data, in this way, we can evaluate the performance with data that are different from the training and validation datasets. Note that the objects of CIFAR100 data are different from CIFAR10 data, thus the evaluation performance will be slightly lower than the results when using CIFAR10 test data for evaluating the performance.

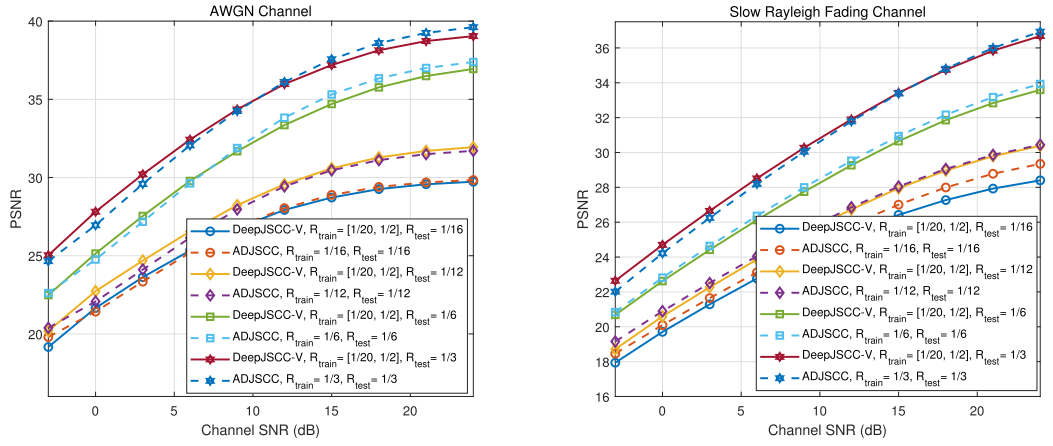
When training the DeepJSCC-V model, in each training epoch we first update the model parameters by using the Adam stochastic optimization method [31] with the CIFAR10 training data and then evaluate the performance of the current model by using the validation dataset, i.e. the CIFAR10 test data. If the MSE performance of the current epoch becomes lower compared with the best MSE performance achieved in previous epochs, we then save the model as the final model to be obtained. The training batch size is set as 128, the learning rate is set as 10^{-4}, and the maximum number of training epochs is set as 400. The training process of OraNet model is similar to DeepJSCC-V, but the difference is that the maximal training epochs are set as 150 since the complexity of OraNet is much lower than DeepJSCC-V.

Except for CIFAR10 and CIFAR100 data, in this paper we will also train the proposed DeepJSCC-V and OraNet models by using the ImageNet data [32], [33] to enable the proposed PADC method can deal with the adaptive transmission problem for large-scale and high-resolution images. To increase the training efficiency and reduce the training samples, we train the DeepJSCC-V model for ImageNet data on the basis of a DeepJSCC-V model pretrained on the CIFAR10 data. We use 45000 images randomly selected from the ImageNet validation dataset as the training data, and use the remained 5000 images as the validation data. For each training sample, it is obtained by center-crop the original images with size 128×128, the batch size is set as 32, and maximal training epochs is set as 50. Other training settings are the same as the training process for CIFAR10 data. After the training process on ImageNet data, we then test the performances of DeepJSCC-V and OraNet on the Kodak24 data [34], which contains 24 color images with size 768×512. To reduce performance fluctuation, all the PSNR performances of each image transmission task are obtained by 100 repeats. After the training process, the accompanied OraNet is trained by using the same ImageNet data, and its training process is the same as the process of the aforementioned OraNet training process on CIFAR10 data.

B. Results on CIFAR Data

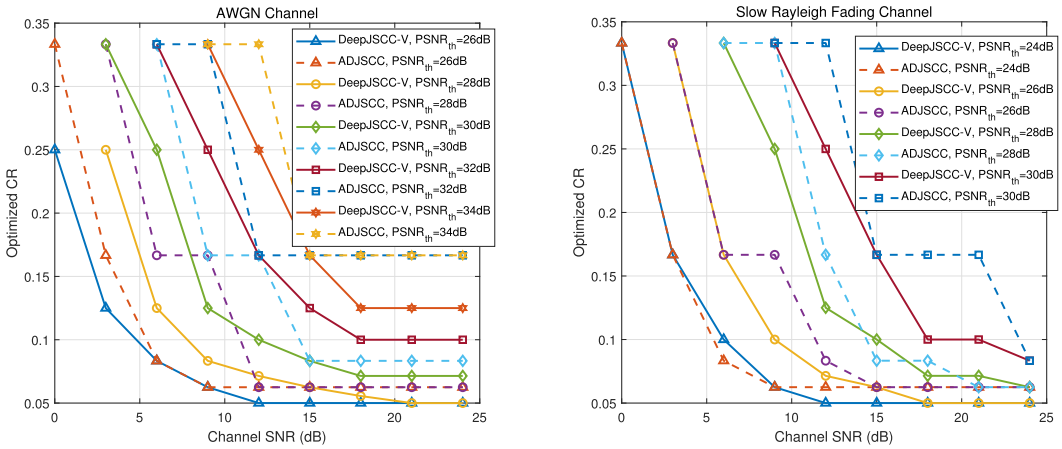
In this subsection, we want to demonstrate that the proposed DeepJSCC-V can achieve better performances on data-level CR optimization compared with the ADJSCC method [23] by using the CIFAR100 test data, and also prove that the proposed OraNet can effectively predict the PSNR values. We won't compare the performance between DeepJSCC-V and DeepJSCC-A proposed in [26] because the Encoder and Decoder networks in DeepJSCC-A are specifically designed for the Policy Network, thus it is not fair to compare the two methods with different network architectures. Most importantly, it is shown that DeepJSCC-A is not able to achieve better performance compared with ADJSCC with the same CR [26]. Therefore, if the proposed DeepJSCC-V can achieve similar PSNR performance compared with ADJSCC, we can know that it outperforms DeepJSCC-A in terms of the PSNR performance with the same CR.

First, we compare the achieved PSNR performances of the proposed DeepJSCC-V and ADJSCC method with the same CR, and Figure 4 shows the results obtained. In this test, we train the DeepJSCC-V model with randomly generated CR values ranging from 0.05 to 0.5. For performance comparison, we train 4 ADJSCC models for each channel type, and the CRs are set as 1/16, 1/12, 1/6, and 1/3. For all the DeepJSCC-V and ADJSCC models, the channel SNR value of each training image data is randomly generated from range $\gamma_{\text{train}} \in [0, 27]$dB. For fairness, except for the SC mask process, the architecture of the encoders and decoders in DeepJSCC-V and ADJSCC are the same. In this way, we can know that the performance differences are caused by the proposed



(a) (b)

Fig. 4. PSNR performance comparison on CIAFR100 image transmission task achieved by the proposed DeepJSCC-V and ADJSCC [23], the training SNRs are randomly generated from $\gamma_{\text{train}} \in [0, 27]$dB, (a) and (b) plot the results achieved in AWGN channel and slow Rayleigh fading channel respectively.



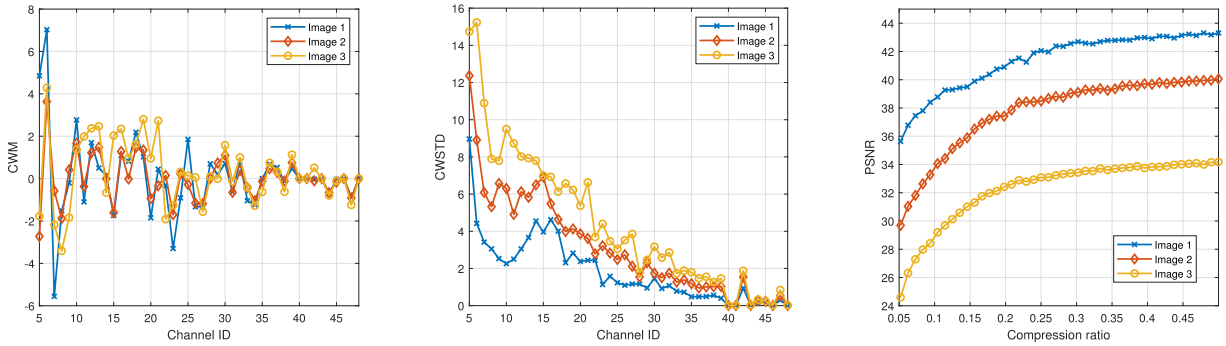
(a) (b)

Fig. 5. Comparison of data-level minimal CR values on CIAFR100 image transmission task achieved by one DeepJSCC-V model and four ADJSCC models [23] with different target PSNR values.

code length adjustment process. In our test, we observe that DeepJSCC-V requires more training epochs to get converged to the best coding performance. For example, ADJSCC may need 150 training epochs to achieve the best capacity, but DeepJSCC-V may need 250 training epochs to achieve the best performance, thus the training cost of DeepJSCC-V is relatively higher than ADJSCC.

From Figure 4(a), we can observe that, with the same CR, the proposed DeepJSCC-V can achieve similar PSNR performances to ADJSCC in both high and low SNR regimes in AWGN channel. In fading channel, as Figure 4(b) shows, we can see that, except for the situation when $R = 1/16$, DeepJSCC-V and ADJSCC achieve similar PSNR performances. However, when CR is very small, i.e. $R = 1/16$, the PSNRs achieved by DeepJSCC-V are slightly lower than ADJSCC, and the performance gap becomes more evident when channel SNR becomes higher. The above results demonstrate that, with only one model, in general, the proposed DeepJSCC-V can achieve similar performance compared with ADJSCC models trained with different CR values, which is helpful for reducing the training and deployment cost. However, when test CRs are relatively low, ADJSCC can achieve slightly better performance compared with the proposed DeepJSCC-V in a slow Rayleigh fading channel.

Next, we test the data-level CR optimization performance achieved by the proposed DeepJSCC-V and ADJSCC. In this test, we only use the DeepJSCC-V model to optimize the data-level CR for all the CIFAR100 images according to the average expected PSNR performance. In this way, the CR values of all images are the same, and OraNet is not necessarily used. We first evaluate the performance of the proposed DeepJSCC-V with different CR and channel SNR conditions. More specifically, we evaluate the achieved PSNR performances with the following 8 CR levels $R \in \{1/16, 1/14, 1/12, 1/10, 1/8, 1/6, 1/4, 1/3\}$. It is necessary to emphasize that, the number of CR levels of DeepJSCC-V can be much larger, such that the model can have more choices when optimizing the code length. For performance comparison, we use the ADJSCC models with CR levels $R \in \{1/16, 1/12, 1/6, 1/3\}$ for providing four different CR choices when optimizing the code length. With the above models, we then test the minimal CR achieved by the proposed DeepJSCC-V and ADJSCC with different channel SNR conditions, and Figure 5 shows the results obtained.



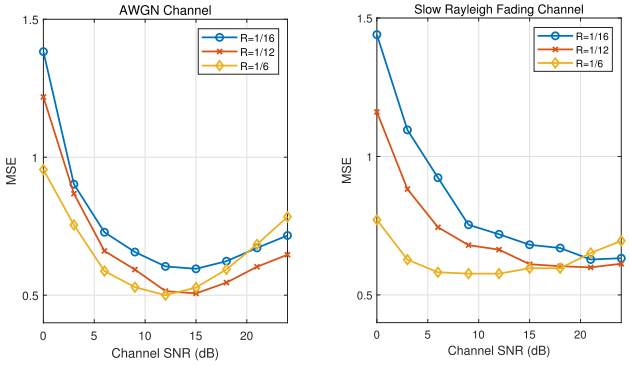
(a) (b) (c)

Fig. 6. Examples of the SC features and the achieved PSNR performances, in which (a)(c)(c) show the channel-wise mean value features, the channel-wise standard deviation features, and the corresponding PSNR performances.

In Figure 5(a), we can observe that, with the same target PSNR, the proposed DeepJSCC-V can achieve much lower CR compared with the ADJSCC method. As aforementioned, in Figure 4(a), DeepJSCC-V can achieve higher PSNR performance compared with ADJSCC in AWGN channels. In addition, even with only one model, in this test DeepJSCC-V can provide 8 CR choices compared with ADJSCC models with only 4 CR choices. Due to the above two benefits, in general, DeepJSCC-V can use much smaller code length for the same image transmission task, thus the overall transmission efficiency will be greatly increased. In Figure 5(b), we can see that the proposed DeepJSCC-V also outperforms ADJSCC in slow Rayleigh fading channel, especially when the target PSNR threshold is large, e.g. $\text{PSNR}^{th} = 30$dB. As we can see in Figure 4(b), with the same channel condition, DeepJSCC-V can achieve similar performance to ADJSCC, thus the main performance improvement is achieved by the flexibility of more CR level choices. In conclusion, we can know that one DeepJSCC-V model with 8 CR choices can achieve much better data-level CR optimization performance compared with the ADJSCC model with 4 choices. Therefore, DeepJSCC-V is able to greatly improve the transmission efficiency compared with ADJSCC, at the same time reduces training and deployment costs.

Before demonstrating the performance of OraNet, we will first show three examples of the CWM and CWSTD features obtained from the SCs, which is helpful for us to understand how DeepJSCC-V works and why the PSNR performance is predictable. For CIFAR data with image size 32×32, the shape of the SC obtained from the Encoder is $48 \times 8 \times 8$. Since we set the CR to be randomly generated from 0.05 to 0.5, thus we can know the corresponding number of channels of MSC ranges from 5 to 48. We randomly select three images from the CIFAR 100 test data, denoted as $\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{x}_3$, and Figure 6 shows the CWM and CWSTD feature of each channel, along with achieved PSNR performance with different CRs, as plotted in (a)(b)(c) respectively.

From Figure 6, we can observe that, for all three images, the absolute values of CWM and CWSTD features gradually decreased from the 5-th channel to 48-th channel, which shows that an inner SC channel carries more information compared with an outer channel with relative larger channel id. This result is similar to the principal component analysis



(a) (b)

Fig. 7. The MSE between the predicted PSNR and ground-truth PSNR, (a) and (b) are the results in AWGN channel and slow Rayleigh fading channel respectively.

(PCA) [35] that decomposes the data as several orthogonal basis, and a basis carries more information if its corresponding eigenvalue is larger. This characteristic is achieved by the SC mask operation, in which we preserve the first K elements in each transmission, this practice will force the Encoder to learn to compress more important information to the inner channels with a relatively smaller number because they are more probable to be transmitted. In this way, the information loss caused by the SC mask process will be minimized.

Denote the PSNRs of the three images $\mathbf{x}_1, \mathbf{x}_1$, and $\mathbf{x}_3$ as PSNR_1, PSNR_2, and PSNR_3 respectively. From Figure 6, we can also see that the PSNR rank is $\text{PSNR}_1 > \text{PSNR}_2 > \text{PSNR}_3$. Accordingly, we can see that, in general, the value ranges of the features are $|\boldsymbol{\mu}_z^{(1)}| < |\boldsymbol{\mu}_z^{(2)}| < |\boldsymbol{\mu}_z^{(3)}|$, and $\boldsymbol{\sigma}_z^{(1)} < \boldsymbol{\sigma}_z^{(2)} < \boldsymbol{\sigma}_z^{(3)}$. Therefore, a relatively larger range of the SC values corresponds to a relatively lower PSNR performance. The reason is that, for an image with more complex contents, the values of CWM and CWSTD features will be relatively larger to carry more information, and accordingly it requires more bandwidth resources to transmit the SC to achieve the same PSNR performance compared with an image that contains less information. These results indicate that the PSNR of the reconstructed image is predictable according to the channel SNR and encoded image features.

Next, we demonstrate that the proposed OraNet can predict high-quality PSNR results for the transmitted images. Figure 7 shows the achieved MSE performances between the

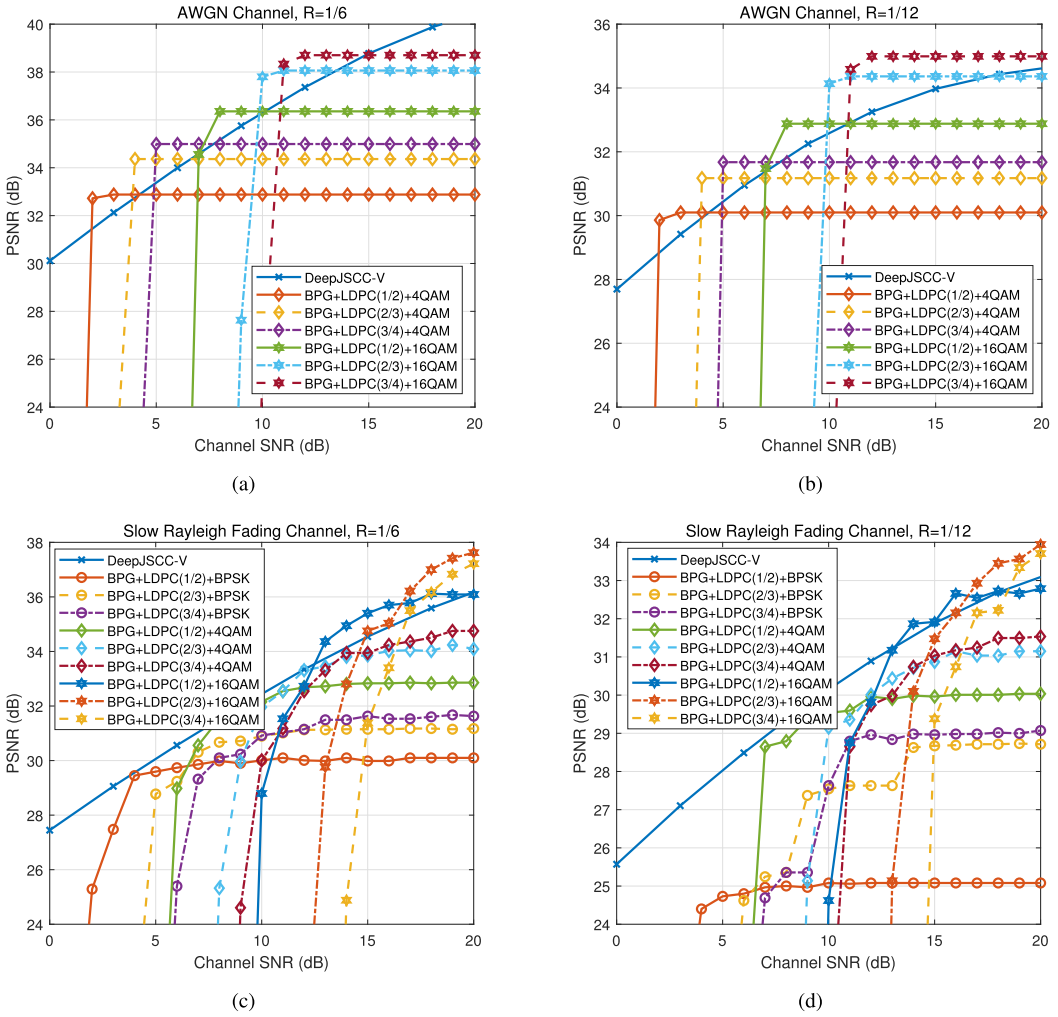


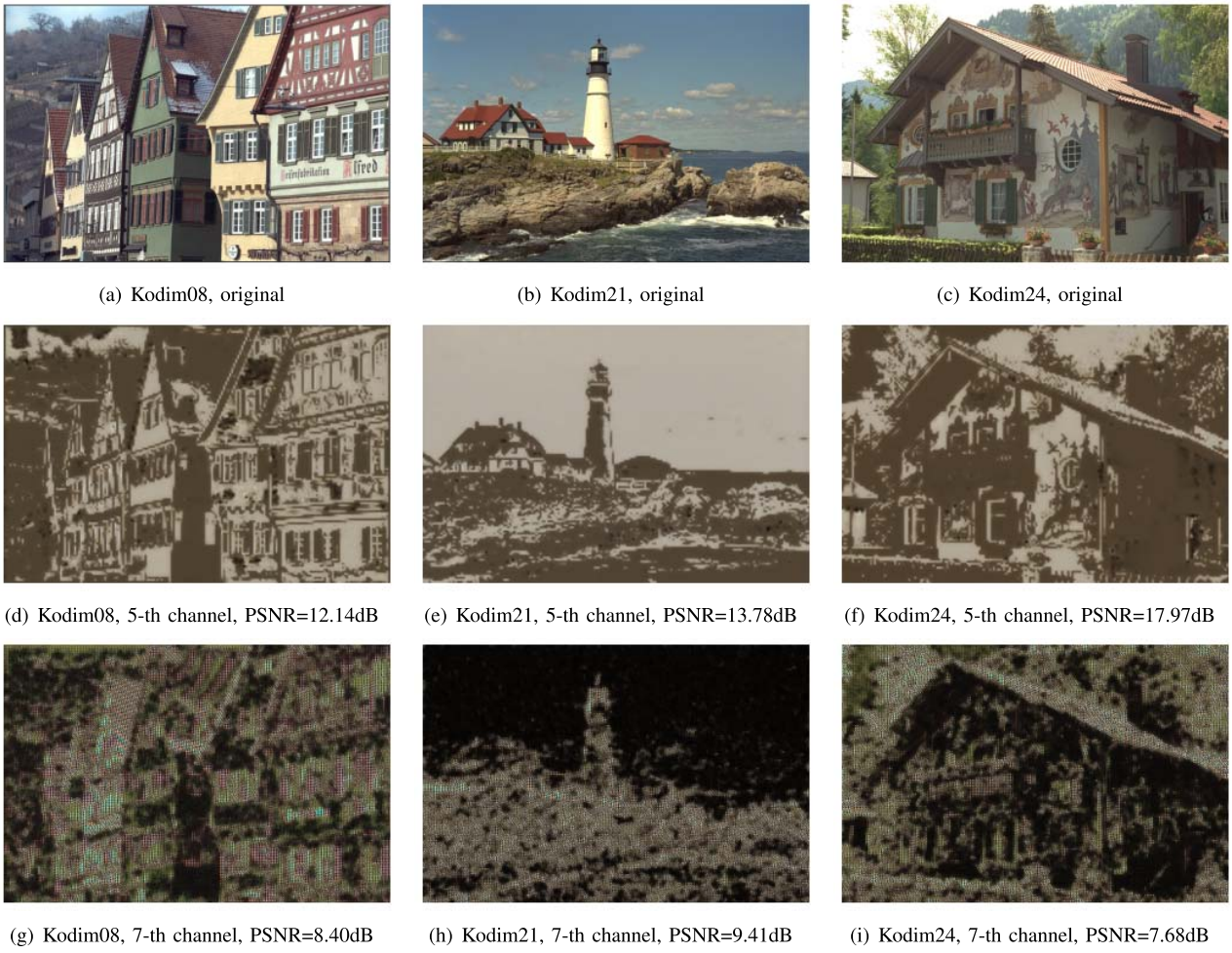
Fig. 8. Performance comparison of DeepJSCC-V and conventional BPG+LDPC transmission schemes, (a)(b) and (c)(d) are the results obtained in AWGN channel and slow Rayleigh fading channel respectively.

predicted PSNR values and the ground-truth PSNR values on the CIFAR100 test data, in (a) and (b) are the results in AWGN channel and slow Rayleigh Fading channel respectively. In general, we can see that when channel SNR is higher than 5dB, the achieved MSE performances can be lower than 1 in both AWGN and slow Rayleigh fading channels, which demonstrated the effectiveness of the proposed OraNet-based PSNR prediction method. However, we can also observe that when channel SNR is very low, e.g. 0dB, the predicting MSE becomes quite large, this is because in a very low SNR regime, the received MSC at the receiver will be heavily corrupted by the stochastic AWGN, causing huge distortion between the original MSC and received MSC, and the PSNR values of will fluctuate in a relative wide value range. Without stable training samples, the OraNet will not be able to predict the PSNR values with high precision. In general, the MSE decreases along with the increase of the channel SNR, but the MSE will be slightly increased when channel SNR keeps increasing in a high SNR regime. The reason is that, when channel SNR is high enough, e.g. higher than 18dB, the achieved PSNR performance of the DeepJSCC-V model will be gradually converged to the upper limit. In this situation, though channel SNR is different, their corresponding PSNR values are almost the same, causing confusion when training the OraNet, and further decreasing the prediction precision. As a result, we can know that the MSE performance has a minimal value, which is achieved when channel SNR is neither too low nor too high.

C. Results on Kodak24 Data

In this subsection, we will test the performance of the proposed PADC method by using the Kodak24 data. Our main goal is to demonstrate that the proposed PADC method can realize adaptive image-level transmission with different channel SNRs and image contents. Some examples will also be provided for understanding the characteristics of the proposed method.

First, we compare the performance of the proposed DeepJSCC-V and conventional BPG+LDPC-based wireless transmission schemes. The BPG codec is a well-known traditional image compression method that can achieve better compression performance compared with methods like JPEG and JPEG2000 [2]. For LDPC, We adopt the channel coding scheme in 802.11n protocol [36], which includes the following three block lengths: 648, 1296, and 1944, and in this test we will fix the code length as 1296. We will test the performance of the BPG+LDPC scheme with the following three code rates: 1/2, 2/3, and 3/4, and the corresponding code combina-



(a) Kodim08, original (b) Kodim21, original (c) Kodim24, original

(d) Kodim08, 5-th channel, PSNR=12.14dB (e) Kodim21, 5-th channel, PSNR=13.78dB (f) Kodim24, 5-th channel, PSNR=17.97dB

(g) Kodim08, 7-th channel, PSNR=8.40dB (h) Kodim21, 7-th channel, PSNR=9.41dB (i) Kodim24, 7-th channel, PSNR=7.68dB

Fig. 9. Examples of the images reconstructed from different SC channels, the channel SNR is set as 12dB.

tions are $(648, 648)$, $(864, 432)$, and $(972, 324)$. Except for the source channel coding methods, the transmission performance is also influenced by the modulation order. In this test, the following three modulation orders will be tested: BPSK, 4QAM, and 16QAM. Note that, due to the 'cliff effect', the transmission performance of conventional transmission schemes may drop sharply when the channel SNR is low, and even fail to transmit the image data. When the achieved PSNRs of conventional wireless transmission schemes are lower than 24 dB, we will not show them in the figure to improve the legibility.

Figure 8 shows the results obtained with different channel cases and channel SNRs. From subfigures (a) and (b), we can observe that, in the AWGN channel case, the PSNR performance of the proposed DeepJSCC-V is relatively lower than BPG+LDPC transmission schemes with different modulation orders, and the performance gap becomes larger when channel SNR becomes higher. From subfigures (c) and (d), we can see that, in the slow Rayleigh fading channel case, DeepJSCC-V can achieve relative better PSNR performance compared with traditional BPG+LDPC schemes in low SNR regime, but relative lower PSNR performance when channel SNR is relative high. These above results are quite similar to the results shown in [9], we can know that except for fading channel with low channel SNR, the proposed DeepJSCC-V model still cannot achieve better PSNR performance compared with the BPG+LDPC schemes in all SNR regime, and in future, we can seek more effective model architectures and training algorithms to further improve the performance of DeepJSCC-V model.

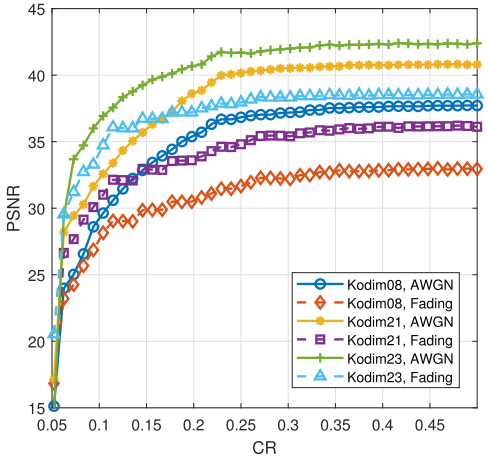
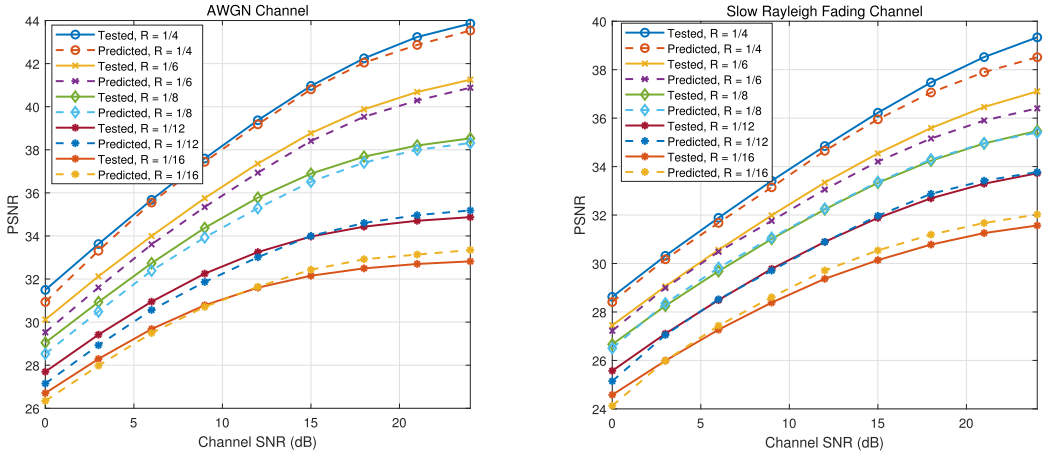


Fig. 10. The PSNR performances on three Kodak24 images achieved by DeepJSCC-V with different CRs.

Next, we want to test the PSNR prediction performance of the OraNet. In Figure 6, we have already shown that the PSNR performance of an image is closely related to the CWM and CWSTD features. To further illustrate the property of the



(a) (b)

Fig. 11. The predicted PSNR and tested PSNR on Kodak24 dataset, (a) and (b) are the results obtained in AWGN channel and slow Rayleigh fading channel respectively.

SC, we use three Kodak images, namely Kodim08, Kodim21, and Kodim24, to show how each SC channel influences the final reconstructed images. Recall that in this paper we set the maximal allowable SC channel number as 48 since we used two downsampling operations in the Encoder, thus the shape of the SC obtained is $48 \times 192 \times 128$. In this test, we set that only one SC channel data can be transmitted to the receiver, and then reconstruct the images with the received one SC channel data. To show the differences of different SC channels, we plot the results obtained from the 5-th and 7-th channels for each image, and Figure 9 shows the results obtained. For the same three images, we further compute the achieved PSNRs with different CRs, the Figure 10 shows the results obtained.

By observing the reconstructed images from 9(d)-(i), we can see that, for the same task, the 5-th channel contains much more image details than the 7-th SC channel. Denote $\text{PSNR}^{(m)}$ as the PSNR performance of the images reconstructed by the m-th SC channel, then we can see the PSNR rank is $\text{PSNR}^{(5)} > \text{PSNR}^{(7)}$. This result shows that an SC channel with a relative inner location contains more image information than other outside SC channels, which is consistent with the results shown in Figure 6. Similar results can be observed in Figure 10, we can see that most of the image information is compressed in the first 20 SC channels, and the information contained by the remained 28 channels is generally quite limited, and this result is also quite similar to the results shown in Figure 6(c). Note that, it doesn't mean that the remaining 28 SC channels are not important, because they contain the image details that are indispensable if we want to obtain very high image quality. Most importantly, when the channel condition is poor, we still need a relative larger CR value to enhance the noise-resilience capability of the channel coding and decoding process.

Next, we evaluate the PSNR performances of the proposed DeepJSCC-V model on Kodak24 data, at the same time compute the corresponding PSNR predictions by using the OraNet model. Figure 11 shows the results obtained, in which (a) and (b) are tested and predicted PSNR results in the AWGN channel and slow Rayleigh fading channel respectively. In this test, we show the results of the following 5 CR levels: 1/16, 1/12, 1/8, 1/6, and 1/4. From the results, we can see that the predicted PSNR values are quite close to the test PSNR values, and in the AWGN channel, the mean absolute value (MAE) between the predicted PSNR value and corresponding tested PSNR values of all the 5 CR levels are 0.293dB, 0.314dB, 0.367dB, 0.423dB, and 0.265dB. In slow Rayleigh fading channel, the MAE of the 5 CR levels are 0.318dB, 0.115dB, 0.0735dB, 0.323dB, and 0.350dB. From the above results, we can see that the OraNet is able to effectively predict the PSNR values with relatively low MAEs.

With the help of OraNet, we then evaluate the performance of the proposed PADC method on the instance-level (or image-level) CR optimization problem. Figure 12 plots the examples of the reconstructed kodim13 and kodim23 images obtained when channel SNR $\gamma = 12$dB and CR $R = 1/8$. Since Kodim13 images contain the forest and river flow objects, and the texture details are much more complex compared with the macaw objects shown in Kodim23, we can know that Kodim13 contains more information, and it requires more bandwidth to transmit the data for the same target PSNR constraint. As a result, we can see that, with the same channel SNR and CR conditions, the PSNR performance of the Kodim13 image is much lower than Kodim23 image, and the PSNR difference between the two images is larger than 8dB, which is very large and cannot be ignored if we want to adjust the data rates with a given target PSNR constraint. This result shows that it is necessary for us to adaptive adjust the data rate according to the different image contents, otherwise, the PSNR performances can fluctuate for different images if we use the same CR value. By using the proposed OraNet, we can see that PSNR performances can be predicted with relatively low error, which enables us to achieve adaptive joint source and channel coding for a specific image under certain channel SNR and image contents.

Figure 13 shows the tested and predicted instance-level CR optimization results for kodim13 and kodim23 images achieved in the AWGN channel and slow Rayleigh fading channel. In this test, we set the available CR choices to

(a) kodim13, AWGN, PSNR=29.95dB, $\widetilde{\text{PSNR}}$=30.15dB (b) kodim23, AWGN, PSNR=38.42dB, $\widetilde{\text{PSNR}}$=37.53dB

(c) kodim13, Fading, PSNR=26.32dB, $\widetilde{\text{PSNR}}$=26.52dB (d) kodim23, Fading, PSNR=35.17dB, $\widetilde{\text{PSNR}}$=35.39dB

Fig. 12. Examples of the kodim13 image and kodim23 images, results in (c)-(f) are all obtained with channel SNR $\gamma = 12$dB, and CR $R = 1/8$, $\widetilde{\text{PSNR}}$ means the predicted PSNR values obtained from the OraNet.

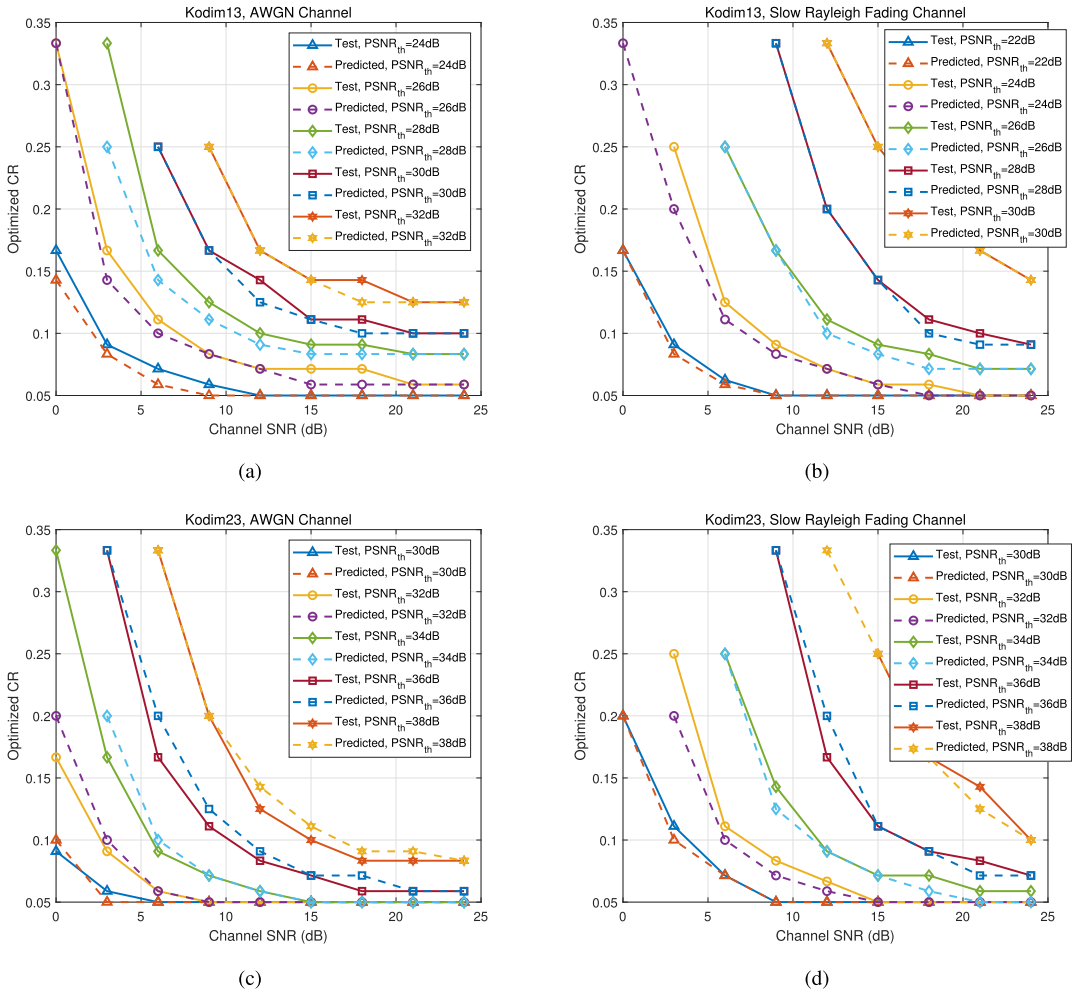


Fig. 13. The predicted PSNR and tested PSNR on Kodim13 and Kodim23 images with different channel SNR and target PSNR values.

have 18 levels, which are $\{1/3, 1/4, \ldots, 1/20\}$, and test the minimal CR values that achieve the target PSNR requirements with different channel SNR conditions. At the same time, we also use the OraNet for predicting the PSNR value of the reconstructed images and predicting the optimal CR value with a given target PSNR value. From the Figure, we can observe that the predicted optimal CR values are very close to the tested ground-truth optimal PSNR values for both high and low target PSNRs, which demonstrates that the proposed PADC method can achieve reasonable CR optimization results for wireless image transmission. We can also see that, a 2dB increase in the target PSNR requirement will cause a significant increase in the CR value in both AWGN and slow Rayleigh fading channels, especially when channel SNR is low. Therefore, to improve the transmission efficiency, it is necessary to reduce the target PSNR value as much as possible, especially in low-SNR environments. We can see that, by using the proposed PADC, the above goal can be effectively achieved.

IV. Conclusion

In this paper, we have proposed a PADC framework to achieve predictive and adaptive code rate control for DeepJSCC-based wireless image transmission. PADC is composed of a DeepJSCC-V model for joint source-channel coding with variable code length, an OraNet model for predicting the PSNR of the image reconstructed by the Decoder by jointly using image contents, channel SNR, and CR, and a CR optimizer for minimizing the CR with a given PSNR constraint. We elaborated the framework of the proposed PADC framework, and the network architectures of the used DeepJSCC-V and OraNet models are also illustrated. Simulation results demonstrated that, with the same channel condition and compression ratio, the proposed DeepJSCC-V model can achieve better or similar performance compared with the state-of-the-art ADJSCC model and also showed that the proposed OraNet model can predict high-quality PSNR values, and the average prediction error of the Kodak24 images are lower than 0.5dB. By jointly using DeepJSCC-V and OraNet, the result proved that the proposed PADC can achieve effective data-level and instance-level code rate optimization with a given target PSNR constraint. Except for data-level CR optimization, PADC can achieve instance-level adaptive rate control for different channel SNRs and image contents, which is helpful for improving the transmission efficiency of DeepJSCC-enabled semantic communications. The simulation results show that, though the used DeepJSCC-V model can achieve similar performance to the ADJSCC model, it still cannot achieve superior performance compared with conventional BPG+LDPC-based transmission schemes in all SNR regime, thus our future work is improving the network architecture and training algorithm of the DeepJSCC-V model to further improve its coding capability.

References

[1] M. W. Marcellin, M. J. Gormish, A. Bilgin, and M. P. Boliek, "An overview of JPEG-2000," in *Proc. Data Compress. Conf.*, 2000, pp. 523–541.

[2] F. Bellard. *BPG Image Format*. Accessed: Jun. 13, 2022. [Online]. Available: http://r0k.us/graphics/kodak/

[3] M. Rabbani and R. Joshi, "An overview of the JPEG 2000 still image compression standard," *Signal Process., Image Commun.*, vol. 17, no. 1, pp. 3–48, Jan. 2002.

[4] Z. Liu, R. Liu, and H. Zhang, "High-throughput adaptive list decoding architecture for polar codes on GPU," *IEEE Trans. Signal Process.*, vol. 70, pp. 878–889, 2022.

[5] H. Vangala, E. Viterbo, and Y. Hong, "A comparative study of polar code constructions for the AWGN channel," 2015, *arXiv:1501.02473*.

[6] S. Weithoffer, C. A. Nour, N. Wehn, C. Douillard, and C. Berrou, "25 years of turbo codes: From Mb/s to beyond 100 Gb/s," in *Proc. IEEE 10th Int. Symp. Turbo Codes Iterative Inf. Process. (ISTC)*, Dec. 2018, pp. 1–6.

[7] G. Shi, Y. Xiao, Y. Li, and X. Xie, "From semantic communication to semantic-aware networking: Model, architecture, and open problems," *IEEE Commun. Mag.*, vol. 59, no. 8, pp. 44–50, Aug. 2021.

[8] S. Kokalj-Filipović and E. Soljanin, "Suppressing the cliff effect in video reproduction quality," *Bell Labs Tech. J.*, vol. 16, no. 4, pp. 171–185, Mar. 2012.

[9] E. Bourtsoulatze, D. B. Kurka, and D. Gunduz, "Deep joint source-channel coding for wireless image transmission," *IEEE Trans. Cognit. Commun. Netw.*, vol. 5, no. 3, pp. 567–579, Sep. 2019.

[10] D. B. Kurka and D. Gunduz, "DeepJSCC-f: Deep joint source-channel coding of images with feedback," *IEEE J. Sel. Areas Inf. Theory*, vol. 1, no. 1, pp. 178–193, Dec. 2020.

[11] M. Gastpar, B. Rimoldi, and M. Vetterli, "To code, or not to code: Lossy source-channel communication revisited," *IEEE Trans. Inf. Theory*, vol. 49, no. 5, pp. 1147–1158, May 2003.

[12] F. Hekland, P. A. Floor, and T. A. Ramstad, "Shannon-Kotel-Nikov mappings in joint source-channel coding," *IEEE Trans. Commun.*, vol. 57, no. 1, pp. 94–105, Jan. 2009.

[13] V. Kostina, Y. Polyanskiy, and S. Verdú, "Joint source-channel coding with feedback," *IEEE Trans. Inf. Theory*, vol. 63, no. 6, pp. 3502–3515, Jun. 2017.

[14] Z. Qin, X. Tao, J. Lu, W. Tong, and G. Ye Li, "Semantic communications: Principles and challenges," 2021, *arXiv:2201.01389*.

[15] P. Zhang et al., "Toward wisdom-evolutionary and primitive-concise 6G: A new paradigm of semantic communication networks," *Engineering*, vol. 8, pp. 60–73, Jan. 2022.

[16] K. Niu et al., "Towards semantic communications: A paradigm shift," 2022, *arXiv:2203.06692*.

[17] N. Farsad, M. Rao, and A. Goldsmith, "Deep learning for joint source-channel coding of text," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 2326–2330.

[18] Z. Weng and Z. Qin, "Semantic communication systems for speech transmission," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 8, pp. 2434–2444, Aug. 2021.

[19] T.-Y. Tung, D. B. Kurka, M. Jankowski, and D. Gündüz, "DeepJSCC-Q: Channel input constrained deep joint source-channel coding," 2021, *arXiv:2111.13042*.

[20] Q. Hu, G. Zhang, Z. Qin, Y. Cai, G. Yu, and G. Ye Li, "Robust semantic communications against semantic noise," 2022, *arXiv:2202.03338*.

[21] D. B. Kurka and D. Gunduz, "Successive refinement of images with deep joint source-channel coding," in *Proc. IEEE 20th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jul. 2019, pp. 1–5.

[22] D. B. Kurka and D. Gunduz, "Bandwidth-agile image transmission with deep joint source-channel coding," *IEEE Trans. Wireless Commun.*, vol. 20, no. 12, pp. 8081–8095, Dec. 2021.

[23] J. Xu, B. Ai, W. Chen, A. Yang, P. Sun, and M. Rodrigues, "Wireless image transmission using deep source channel coding with attention modules," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 4, pp. 2315–2328, Apr. 2022.

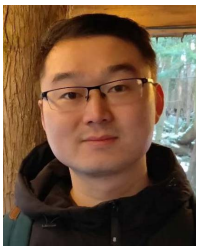
[24] K. Sayood, H. H. Otu, and N. Demir, "Joint source/channel coding for variable length codes," *IEEE Trans. Commun.*, vol. 48, no. 5, pp. 787–794, May 2000.

[25] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

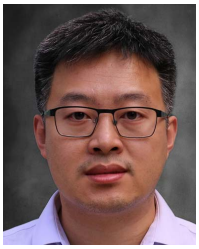
[26] M. Yang and H.-S. Kim, "Deep joint source-channel coding for wireless image transmission with adaptive rate control," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2022, pp. 5193–5197.

[27] Y. Polyanskiy, H. V. Poor, and S. Verdú, "Channel coding rate in the finite blocklength regime," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2307–2359, May 2010.

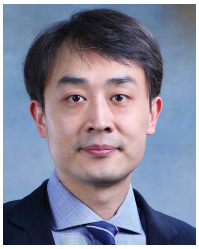
[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[29] J. Ballé, V. Laparra, and E. P. Simoncelli, "Density modeling of images using a generalized normalization transformation," 2015, *arXiv:1511.06281*.
[30] A. Krizhevsky. *CIFAR Dataset*. Accessed: Mar. 18, 2021. [Online]. Available: https://www.cs.toronto.edu/~kriz/cifar.html
[31] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[32] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[33] O. Russakovsky et al., "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.
[34] R. Franzen. *kodak24 Dataset*. kodak24 dataset. [Online]. Available: http://r0k.us/graphics/kodak/
[35] H. Abdi and L. J. Williams, "Principal component analysis," *WIREs Comput. Statistic*, vol. 2, no. 4, pp. 433–459, Jul./Aug. 2010.
[36] A. Mahdi, N. Kanistras, and V. Paliouras, "A multirate fully parallel LDPC encoder for the IEEE 802.11 n/AC/ax QC-LDPC codes based on reduced complexity XOR trees," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 29, no. 1, pp. 51–64, Jan. 2021.



Wenyu Zhang received the B.E. and M.Eng. degrees in communication engineering from Beijing Jiaotong University, Beijing, China, in 2013 and 2016, respectively, the joint Ph.D. degree from The University of British of Columbia in 2019, and the Ph.D. degree in communication engineering from Beijing Jiaotong University in 2020. He is currently an Associate Professor with the School of Intelligence Science and Technology, Institute of Artificial Intelligence, University of Science and Technology Beijing. His research interests include network intelligence and cloud/edge computing.



Haijun Zhang (Fellow, IEEE) is currently a Full Professor and an Associate Dean of the School of Computer and Communications Engineering, University of Science and Technology Beijing, China. He was a Post-Doctoral Research Fellow with the Department of Electrical and Computer Engineering, The University of British Columbia (UBC), Canada. He serves/served as the Track Co-Chair for VTC Fall 2022 and WCNC 2020/2021, the Symposium Chair for Globecom'19, the TPC Co-Chair for INFOCOM 2018 Workshop on Integrating Edge Computing, Caching, and Offloading in Next Generation Networks, and the General Co-Chair for GameNets'16. He serves/served as an Editor for IEEE TRANSACTIONS ON COMMUNICATIONS and IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING. He received the IEEE CSIM Technical Committee Best Journal Paper Award in 2018, IEEE ComSoc Young Author Best Paper Award in 2017, and IEEE ComSoc Asia-Pacific Best Young Researcher Award in 2019.



Hui Ma received the Ph.D. degree in electrical engineering from The University of British Columbia in 2019. He is currently a Lecturer with the School of Intelligence Science and Technology, University of Science and Technology Beijing, China. His research interests include intelligent communications, collaborative communications, massive MIMO, physical layer security, and heterogeneous networks.

Hua Shao received the Ph.D. degree from BUPT in 2017. Since 2017, he has been a Research Engineer with the Wireless Research Department, Huawei Chengdu Research Center, focusing on 5G physical layer techniques and 3GPP standards progressing. In 2020, he enrolled at the School of Intelligence Science and Technology, Institute of Artificial Intelligence, USTB. His research interests include high-frequency physical layer techniques, positioning, sparse signal processing, and AI approaches to reform the RAN physical layer modules.

Ning Wang (Member, IEEE) received the B.E. degree in communication engineering from Tianjin University, China, in 2004, the M.A.Sc. degree in electrical engineering from The University of British Columbia, Canada, in 2010, and the Ph.D. degree in electrical engineering from the University of Victoria, Canada, in 2013. From 2004 to 2008, he was with the China Information Technology Consulting and Designing Institute as a Mobile Communication System Engineer, specializing in planning and optimization of commercial mobile communication networks. From 2013 to 2015, he was a Post-Doctoral Research Fellow with the Department of Electrical and Computer Engineering and the Institute for Computing, Information and Cognitive Systems (ICICS), The University of British Columbia. Since 2015, he has been with the School of Information Engineering, Zhengzhou University, Zhengzhou, China, where he is currently a Professor and the Director of the Henan International Joint Laboratory for Intelligent Networking and Data Analysis. He also holds adjunct appointment with the Department of Electrical and Computer Engineering, McMaster University, Canada. His research interests include resource allocation and security designs of future cellular networks, channel modeling for wireless communications, statistical signal processing, and cooperative wireless communications. He was on the technical program committees of international conferences, including the IEEE GLOBECOM, IEEE ICC, IEEE WCNC, and CyberC.

Victor C. M. Leung (Life Fellow, IEEE) is currently a Distinguished Professor in computer science and software engineering at Shenzhen University, China. He is also an Emeritus Professor in electrical and computer engineering and the Director of the Laboratory for Wireless Networks and Mobile Systems, The University of British Columbia (UBC), Canada. His research interests include the broad areas of wireless networks and mobile systems, and he has published widely in these areas. He is serving on the editorial boards of the IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING, IEEE TRANSACTIONS ON CLOUD COMPUTING, IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS, IEEE ACCESS, *IEEE Network*, and several other journals. He received the 1977 APEBC Gold Medal, NSERC Postgraduate Scholarships (1977–1981), IEEE Vancouver Section Centennial Award, 2011 UBC Killam Research Prize, 2017 Canadian Award for Telecommunications Research, 2018 IEEE TCGCC Distinguished Technical Achievement Recognition Award, and 2018 ACM MSWiM Reginald Fessenden Award. He has coauthored papers that won the 2017 IEEE ComSoc Fred W. Ellersick Prize, 2017 IEEE Systems Journal Best Paper Award, 2018 IEEE CSIM Best Journal Paper Award, and 2019 IEEE TCGCC Best Journal Paper Award. He is a fellow of the Royal Society of Canada (Academy of Science), Canadian Academy of Engineering, and Engineering Institute of Canada. He is named in the current Clarivate Analytics list of "Highly Cited Researchers."