Deep Joint Source-Channel Coding for Wireless Image Transmission with Entropy-Aware Adaptive Rate Control

Weixuan Chen*, Yuhao Chen[†], Qianqian Yang^{*§}, Chongwen Huang^{*}, Qian Wang[‡], Zhaoyang Zhang^{*}

College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310007, China

[†] College of Control Science and Engineering, Zhejiang University, Hangzhou 310007, China

[‡] College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China

{12231075, csechenyh, [§]qianqianyang20, chongwenhuang, ning_ming}@zju.edu.cn

wangqian18@zjut.edu.cn

Abstract-Adaptive rate control for deep joint source and channel coding (JSCC) is considered as an effective approach to transmit sufficient information in scenarios with limited communication resources. We propose a deep JSCC scheme for wireless image transmission with entropy-aware adaptive rate control, using a single deep neural network to support multiple rates and automatically adjust the rate based on the feature maps of the input image and their entropy, as well as the channel conditions. In particular, we maximize the entropy of the feature maps to increase the average information carried by each transmitted symbol during the training. We further decide which feature maps should be activated based on their entropy, which improves the efficiency of the transmitted symbols. We also propose a pruning module to remove less important pixels in the activated feature maps in order to further improve transmission efficiency. The experimental results demonstrate that our proposed scheme learns an effective rate control strategy that reduces the required channel bandwidth while preserving the quality of the reconstructed images.

Index Terms—Joint source and channel coding, adaptive rate control.

I. INTRODUCTION

Conventional wireless image transmission systems typically perform source coding and channel coding of the source image separately. Source coding first compresses the source image to remove redundant information. Then, channel coding adds redundant bits to the source-coded image to ensure reliable wireless transmission. According to Shannon's separation theorem [1], this structure requires the transmission of infinitely long codewords to achieve theoretical optimality. However, in practical scenarios such as autonomous driving and telemedicine, long code block lengths are difficult to achieve. Therefore, separation-based coding schemes may not be optimal for wireless transmission.

To address this issue, deep learning techniques have been introduced to wireless image transmission systems to offer significant performance improvements. Bourtsoulatze et al. proposed the joint source and channel coding (JSCC) algorithm for wireless image transmission based on deep learning [2], which is referred to as the deep JSCC scheme. In this scheme, the authors train a deep learning model to extract a number of feature maps from the source image, with the non-trainable communication channel incorporated into the training process. Experimental results have shown that the deep JSCC scheme achieves better performance than the separation-based digital transmission scheme. Based on the deep JSCC scheme, several studies have been conducted to further improve its performance. Kurka et al. [3] proposed DeepJSCC-f, which introduces the channel output feedback to the deep JSCC scheme. This achieves variable-length transmission and reduces the required channel bandwidth. Zhang et al. [4] proposed MLSC-image, a multi-level semantic communication system for image transmission. It extracts both high-level and low-level image semantic features to improve image reconstruction performance.

However, there are still some aspects of the pioneering deep JSCC scheme that could be improved. On one hand, deep JSCC maps the pixel values of the input image to the complex-valued channel input symbols, treating each symbol equally with the same channel resources. However, for different transmission purpose, the transmitted symbols may have different importance. On the other hand, the transmission rate of the deep JSCC scheme is fixed, which limits the transmission of complex image content when communication resources are limited. To address this limitation, the deep JSCC scheme could be made more flexible and effective by dynamically adjusting the transmission rate to transmit important information with limited channel bandwidth.

Regarding the first aspect, we believe that symbols from feature maps with higher entropy are more important because they have more information on average and thus may contribute to better image reconstruction. As for the second aspect, several studies on multi-rate deep JSCC schemes have been conducted. Kurka et al. [5] [6] studied the problem of adaptive bandwidth image transmission over wireless channels. Yang et al. [7] supported multiple rates using a single deep neural network that can dynamically adjust its rate based on the channel signal-to-noise ratio (SNR) and the feature maps. However, these studies only considered reducing the number of feature maps to be transmitted, ignoring the fact that the length of each feature map can also be reduced. In contrast, Zhou et al. [8] proposed a multi-bit length semantic encoding scheme that can reasonably decrease the length of transmitted information bits and minimize the overhead required for the correct delivery of information. Inspired by [8], we believe that not only some feature maps may be unimportant, but even some of the pixels within an important feature map are also unimportant.

In this paper, we propose an entropy-aware deep JSCC scheme that can automatically adjust its rate based on the feature maps and their entropy, as well as the channel conditions. Specifically, we define the feature maps that need to be transmitted as the **activated feature maps**. We select the activated feature maps from all feature maps of the image and then prune them to increase the flexibility of the transmission rate. The main contributions of this paper are as follows:

• Entropy-Aware: We maximize the entropy of the feature maps during training to increase the average amount of information carried by each transmitted symbol and determine the activated feature maps based on their entropy. This benefits the image reconstruction performance of the model.

• **Sparse Feature Maps**: We prune the activated feature maps while maintaining their structural information. This reduces the transmission of unimportant symbols, resulting in reduced channel bandwidth usage.

• Automatic Rate Adaption: The selection and pruning of feature maps are fully automatic. We use two policy networks to determine the activated feature maps and the pruning ratio for these activated feature maps, respectively.

Our experimental results demonstrate that the proposed method is effective in selecting the activated feature maps and pruning them under different channel conditions. Compared to existing methods, our proposed method achieves a notable improvement in image reconstruction performance, with a PSNR increase of approximately 0.11-0.57dB.

II. OUR PROPOSED DEEP JSCC SCHEME WITH ENTROPY-AWARE ADAPTIVE RATE CONTROL

A. Overall Structure

In this section, we propose a deep JSCC scheme that can automatically adjust its rate based on the feature maps and their entropy, as well as the channel SNR, while maintaining high image reconstruction performance. The proposed scheme consists of several components, namely a semantic encoder, a feature map selection and pruning module, a channel encoder, a communication channel, a channel decoder, and a semantic decoder.

To transmit an image x, we feed it into the semantic encoder for extracting feature map $z = f_{se}(x)$. Then in the feature map selection and pruning module, both z and its entropy H_z are provided as inputs to a policy network P_1 in order to generate a mask M, based on which the activated feature map z_1 is selected. z_1 is then input to another policy network P_2 to obtain its pruning ratio, which is applied by the feature map pruning module to prune z_1 , resulting in the pruned feature map z_2 and the pruning index matrix M_p . After obtaining z_2 and M_p , they are fed into the channel encoder to yield a complex-valued transmitted signal, represented as z_3 . z_3 is passed through the communication channel and becomes \hat{z}_3 . \hat{z}_3 contains two components: the received signal \hat{z}_2 and the received modulated pruning index matrix \hat{M}'_p . At the receiver, the channel decoder maps \hat{z}_2 back to \hat{z} according to \hat{M}'_p , from which the semantic decoder then reconstructs the image \hat{x} . A comprehensive illustration of our proposed deep JSCC scheme is presented in Fig. 1. Next, we will introduce each module in detail.

B. Semantic Encoder and Decoder

The semantic encoder takes x as input and outputs the feature map z, while the semantic decoder reconstructs the image from \hat{z} . The semantic encoder consists of two components, S_1 and S_2 . The semantic encoder S_1 consists of three convolutional layers, while semantic encoder S_2 includes two ResNet blocks, two SNR adaptive modules, and one convolutional layer. The structure of the SNR adaptive module is identical to that in [7]. The structure of the semantic encoder we also note that some convolutional layers in the semantic encoder become transpose convolutional layers for upsampling in the semantic decoder.

C. Feature Map Selection and Pruning Module

The feature map selection and pruning module selects the activated feature maps and prunes them. Firstly, we define the dimension of z as $2C \times \frac{L}{2}$, where 2C represents the number of feature maps and $\frac{L}{2}$ is the length of each feature map. We calculate the entropy of each feature map z_i as $H(z_i)$, where i is the order index. Then, in order to increase the difference between the entropy values of the feature maps, we apply the softmax function $H'(z_i) = \frac{e^{H(z_i)}}{\sum_{i=1}^{2C} e^{H(z_i)}}$ to normalize $H(z_i)$. The overall entropy matrix H_z is obtained by concatenating

The overall entropy matrix H_z is obtained by concatenating all $H'(z_i)$. Then every two feature maps in z are concatenated to form a concatenated feature map $z' \in \mathbb{R}^{C \times L}$, since the feature maps are input to the communication channel with one half as the real part and the other half as the imaginary part. After concatenation, all C feature maps can be selected as activated feature maps since this can further increase the flexibility of our transmission strategy. However, only some of the feature maps were selective in the previous method [7].

Next, we feed z and H_z into a policy network P_1 . This network outputs a binary mask $M \in \{0, 1\}^{C \times 1}$ that indicates which feature maps should be activated. P_1 learns to select the activated feature maps based on all the feature maps and their entropy, as well as the channel SNR, whose structure is shown in Fig. 2. As shown in Fig. 2, we first concatenate z and H_z to construct an information matrix for the feature maps. We then average each row of this matrix and concatenate the resulting matrix with the channel SNR. This generates a $(2C + 1) \times 1$ matrix. The probabilities for each possible selection are generated by passing this matrix into a twolayer multi-layer perceptron (MLP). To overcome the non-



Fig. 1: The overall structure of our proposed deep JSCC scheme.



Fig. 2: The structure of policy network P_1 .

differentiability problem caused by the discrete nature of the sampling process, we utilize Gumbel-Softmax to sample the selection as a one-hot vector. This vector is then transformed into the mask M. The adaptive transmission mask has a length of C, where each "1" in the mask means that the feature map at the corresponding position in z' will be activated, and vice versa. We then multiply the concatenated feature map z' by M to obtain the activated feature map z_1 . Note that the inputs of P_1 are z and H_z , while the activated feature maps that are not activated are zero-padded. We denote the total number of



Fig. 3: The structure of policy network P_2 .

activated feature maps as $\hat{C} = \sum_{i=1}^{C} M_i$.

In the next step, we use another policy network P_2 to determine the pruning ratio for the activated feature maps. P_2 takes z_1 as its input and outputs a one-hot vector V of length T, where the T denotes the number of possible pruning ratios. Its structure is shown in Fig. 3. As shown in Fig. 3, we first average each row of z_1 and concatenate the resulting matrix with the channel SNR. The concatenated matrix is then fed into the MLP, which consists of two fully connected layers and a softmax function at the end. By utilizing the maximum probability value, we generate the one-hot vector of length T. In this paper, we set T to 5 and define the possible pruning ratios as $a \in \{0, 0.2, 0.25, 0.3, 0.35\}$. Once we obtain the onehot vector V, we apply pruning to z_1 based on V to obtain the pruned feature map z_2 . The dimension of z_2 is $\hat{C} \times \hat{L}$. Specifically, inspired by the commonly used l1-norm based pruning rule in model pruning [9], for each activated feature map, we regard its pixel with smaller l1-norm (i.e., pixels with smaller absolute numerical values) as being less important. During the pruning process, we remove a portion of pixels with relatively small *l*1-norm in each activated feature map according to the pruning ratio obtained from V. However, the positions of these removed pixels in the activated feature maps contain important structural information of the input image. Therefore, we record the position indices of the pruned pixels in a pruning index matrix $M_{\rm p} \in \{0,1\}^{\hat{C} \times L}$, where the "1" indicates that the pixel at that position has been pruned. It is essential to transmit $M_{\rm p}$ to the receiver so that the receiver can zero-pad the pruned pixels according to $M_{\rm p}$, thus recover the structure of the activated feature maps.

D. Channel Encoder and Decoder

In the channel encoder, we use 64-quadrature amplitude modulation (64-QAM) to modulate $M_{\rm p}$, resulting in a length of each vector of $L' = \left\lceil \frac{L}{6} \right\rceil$. Then, we obtain the modulated pruning index matrix $M'_{\rm p}$ for the activated feature maps. It has a dimension of $\hat{C} \times L'$ pixels. We input the pruned feature map z_2 and the modulated pruning index matrix $M'_{\rm p}$ into the power normalization module, which generates a complexvalued transmitted signal $z_3 \in \mathbb{C}^{\hat{C} \times (\frac{\hat{L}+L'}{2})}$ with unit average power. After that, we input z_3 into the noisy wireless channel and output \hat{z}_3 . \hat{z}_3 contains the received signal \hat{z}_2 and the received modulated pruning index matrix $\hat{M}'_{\rm p}$. Finally, we feed \hat{z}_2 and $\hat{M}'_{\rm p}$ into the channel decoder.

In the channel decoder, we demodulate $\hat{M}'_{\rm p}$ and zero-pad the pruned pixels according to $\hat{M}_{\rm p}$ to recover the structure of the activated feature maps. Subsequently, we zero-pad the feature maps that are not activated. The output of the channel decoder is \hat{z} .

E. Communication Channel

The communication channel is modeled as a non-trainable layer that adds random perturbation to the transmitted symbols. In this paper, we consider AWGN wireless channel. The output of the channel can be expressed as $\hat{z}_3 = z_3 + n$, where *n* represents the additive Gaussian noise. The channel SNR, which determines the level of noise in the channel, is known by both the transmitter and receiver, allowing the semantic encoder, semantic decoder, P_1 , and P_2 to adapt to different channel conditions.

F. Transmission Rate

Our objective is to maintain the quality of reconstructed image while reducing the channel bandwidth usage. We measure the transmission rate based on the wireless channel usage per pixel (CPP). If the input image has a dimension of $3 \times H \times W$ pixels, then CPP = $\frac{\hat{C}(\hat{L}+L')}{2HW}$. Recall that \hat{C} represents the number of activated feature maps, \hat{L} represents the length of the pruned activated feature maps, and L' represents the number of columns of the modulated pruning index matrix $M'_{\rm p}$. When no feature maps are pruned, L' = 0. Hence, the value of CPP mainly depends on the number and length of the activated feature maps, as well as the number of pixels removed from the activated feature maps.

G. Loss Function and Training Strategy

To train the proposed deep JSCC scheme, we incorporate three terms into our loss function \mathcal{L} , which is defined as :

$$\mathcal{L} = \mathbb{E}\left[\left\| x - \hat{x} \right\|_{2}^{2} + \alpha \frac{\hat{L}}{L} \sum_{i=1}^{C} M_{i} - \beta \mathbb{E}\left(e^{H(z_{i})} \right) \right].$$
(1)

Here, our loss function takes the expected value over the entire training set. The first term is the mean square error between the input image and the reconstructed image, ensuring image reconstruction performance. The second term is the product of the length ratio and the number of the activated feature maps. This term represents wireless channel usage and helps reduce the required channel bandwidth. The third term indicates the average processed entropy of z, which increases the average amount of information carried by each transmitted symbol. It is calculated by taking the expected value of the processed entropy of all the feature maps. Note that here we do not normalize these entropy values. We use two hyperparameters α and β to balance these three terms.

To train our model, we use the Adam optimizer [10]. Specifically, we first train the entire model for 200 epochs with a learning rate of 5×10^{-4} . Afterward, we decrease the learning rate to 5×10^{-5} and continue training for another 200 epochs. Afterwards, we proceed to the fine-tuning phase, where the learning rate becomes 1×10^{-5} . During this phase, we first freeze the semantic encoder S_1 and train the model for 100 epochs. Then, we freeze the semantic encoder S_2 and the policy networks, and train the model for 100 epochs.

III. EXPERIMENTS

A. Experimental Setup

We evaluate our proposed scheme using the CIFAR-10 dataset, which consists of 50,000 training images and 10,000 testing images. All images are 32×32 pixel RGB images. The batch size is set to 512, and the channel SNR is uniformly sampled between 0dB and 15dB for each image in every epoch during the training process. The hyperparameter α_{ours} has two distinct values: 2×10^{-4} and 1.3×10^{-3} . β is set to 1×10^{-5} . The number of filters in the last layer of the semantic encoder is 2C = 16, and the length of the corresponding feature maps is $\frac{L}{2} = 64$. The feature maps in z are then concatenated in pairs to create selective feature map z' with dimension $C \times L = 8 \times 128$. If the feature maps are pruned, the binary vectors used to represent the pruning indices comprise 128 bits. Thus, $L' = \left\lceil \frac{128}{6} \right\rceil = 22$.

TABLE I: The strategy comparison between the baseline [7] with $\alpha_{\text{base}} = 5 \times 10^{-4}$ and our method with $\alpha_{\text{ours}} = 2 \times 10^{-4}$.

SNR(dB)	Average number of feature maps	Average feature map length ratio(%)	Rate(CPP)	Baseline Rate(CPP)	PSNR(dB)	Basline PSNR(dB)
15	7.90	74.04	0.450	0.447	33.37 (↑0.57)	32.80
10	7.97	100.00	0.498	0.488	31.79 (↑0.59)	31.20
5	8.00	100.00	0.500	0.499	28.60 (†0.28)	28.32
0	8.00	100.00	0.500	0.500	24.74 (†0.21)	24.53

TABLE II: The strategy comparison between the baseline [7] with $\alpha_{\text{base}} = 1.5 \times 10^{-3}$ and our method with $\alpha_{\text{ours}} = 1.3 \times 10^{-3}$.

SNR(dB)	Average number of feature maps	Average feature map length ratio(%)	Rate(CPP)	Baseline Rate(CPP)	PSNR(dB)	Basline PSNR(dB)
15	4.60	73.20	0.260	0.255	30.06(↑0.11)	29.95
10	5.54	100.00	0.346	0.310	29.54(↑0.65)	28.89
5	7.29	100.00	0.455	0.430	27.88(↑0.40)	27.48
0	7.90	100.00	0.493	0.493	24.56(↑0.11)	24.45



Fig. 4: The comparison of image reconstruction performance between our method, the method proposed in [7], and the BPG+capacity method.

For comparison, we use two baseline methods: the adaptive rate control scheme proposed by Yang et al. [7] and the BPG image codec combined with idealistic error-free transmission based on Shannon capacity. We use peak signal-to-noise ratio (PSNR) as the performance metric.

B. Experimental Results

First, we analyze the number of feature maps and the pruning ratios selected by our proposed method under different SNRs. For comparison, we use data obtained from the baseline [7], which is shown in Table. I and Table. II. To ensure a fair performance comparison, we set α_{ours} in multiple attempts to maintain CPP levels similar. When the SNR is low (0-10dB), our method selects a larger number of feature maps and we do not prune them in this case. This is because poor channel conditions require more feature maps to be transmitted to provide enough information for image reconstruction. Pruning is not worthwhile in this case because 64-QAM has a high bit error rate (BER) at low SNRs. Our method achieves approximately 0.11-0.28dB higher PSNR compared to the baseline without pruning the feature maps. This indicates that our designed entropy-aware feature map selection module can help improve image reconstruction performance. As the SNR increases to 15dB, we prune about 26%-27% of the pixels in the activated feature maps.

Next, we compare our proposed method with the method proposed by Yang et al. [7] and the BPG+capacity method.

We compare the performance of these models and plot the SNR-PSNR curves, as shown in Fig. 4. Our model outperforms the method proposed in [7] for both $\alpha_{ours} = 2 \times 10^{-4}$ and $\alpha_{ours} = 1.3 \times 10^{-3}$. At similar CPPs, our method improves PSNR by about 0.21-0.57dB when α_{ours} is relatively small. However, the PSNR improvement drops to around 0.11dB when we increase the value of α_{ours} . This is because that as α_{ours} increases, the policy networks focus more on channel bandwidth usage rather than PSNR performance, thereby reducing the effectiveness of our scheme. At low SNRs (0-10dB), our method outperforms both baseline methods. At high SNR (15dB), our method performs equivalently to the BPG+capacity method when α_{ours} is small, while completely outperforming it when α_{ours} increases. Note that the BPG+capacity method is rate-fixed. Therefore, even if the PSNR performances are similar, our method has the advantage of rate adaptivity. Compared to the method proposed in [7], our method shows a notable improvement in image reconstruction performance, achieving approximately 0.57dB for small α_{ours} and slightly lower, about 0.11dB, for larger α_{ours} . This shows that our method is more advantageous when channel resources are sufficient.

Finally, we consider two types of images in CIFAR-10 transmitted by our proposed scheme, as shown in Fig. 5. In Fig. 5, the blue bar represents the total entropy of the activated feature maps, while the red curve represents the entropy of the input images, both of which refer to the average. The



Fig. 5: The comparison of transmission schemes for two types of images with different complexity in CIFAR-10.

TABLE III: The performance comparison of the proposed

method with and without the feature map pruning technique. $\frac{Model}{\alpha_{ours} = 2 \times 10^{-4} \text{ (with } P_2)} = 0.450 \quad 33.37(\uparrow 0.18) \\ \alpha_{ours} = 4.5 \times 10^{-4} \text{ (without } P_2) = 0.451 \quad 33.19$

$\alpha_{\rm ours} = 2 \times 10^{-4}$ (with P_2)	0.450	33.37(^0.18)
$\alpha_{\rm ours} = 4.5 \times 10^{-4}$ (without P_2)	0.451	33.19
$\alpha_{\rm ours} = 3.5 \times 10^{-4}$ (with P_2)	0.433	33.10(\cap 0.16)
$\alpha_{\rm ours} = 4.8 \times 10^{-4}$ (without P_2)	0.433	32.94
$\alpha_{\rm ours} = 4 \times 10^{-4}$ (with P_2)	0.406	32.65(\cprectrime 0.09)
$\alpha_{\rm ours} = 5.4 \times 10^{-4}$ (without P_2)	0.405	32.56
$\alpha_{\rm ours} = 5.8 \times 10^{-4}$ (with P_2)	0.356	31.77(\phi0.02)
$\alpha_{\rm ours} = 7.5 \times 10^{-4}$ (without P_2)	0.356	31.75
$\alpha_{\rm ours} = 1.3 \times 10^{-3} \text{ (with } P_2\text{)}$	0.260	30.06(↑0.02)
$\alpha_{\rm ours} = 1.9 \times 10^{-3}$ (without P_2)	0.260	30.04

two types of images have different complexity, with higher entropy representing higher complexity. For each type of image, we take the average of 100 images. For the simple images, our method selects an average of 4.11 feature maps and prunes 28.2% pixels of them, resulting in an average CPP of 0.229. While for the complex images, our method selects an average of 4.83 feature maps and prunes 27.9% pixels of them, resulting in an average CPP of 0.269. The result shows that our method selects fewer feature maps and therefore lower CPP when the image content is relatively simple, and vice versa. It also proves that our method can automatically select activated feature maps based on the entropy of the image. In conclusion, by combining the entropy-aware feature map selection with the feature map pruning, we achieve notable improvement in image reconstruction performance.

C. Ablation Study

We also conduct an ablation study of the proposed scheme to evaluate its effectiveness, as shown in Table. III. Specifically, we train several models with different α_{ours} values to compare the performance of models with and without the feature map pruning technique. To ensure a fair performance comparison, we set the α_{ours} values through multiple attempts to ensure similar CPP levels.

By comparing the results, we find that pruning the activated feature maps can improve image reconstruction performance of the model with similar CPPs. The effect of the feature map pruning technique is more significant at lower α_{ours} . These experiments confirm that our feature map pruning technique effectively improves the image reconstruction performance.

IV. CONCLUSION

In this paper, we proposed a deep JSCC scheme for wireless image transmission with entropy-aware adaptive rate control. During training, we maximized the entropy of the feature maps to increase the amount of average information carried by each transmitted symbol, and selected activated feature maps based on their entropy. This helped the model to achieve better image reconstruction performance. To achieve the adaptive transmission rate, we introduced two policy networks. The first generated a binary mask that selects activated feature maps. Subsequently, the second policy network decided the pruning ratio for the activated feature maps. We transmitted the position information of the pruned pixels over the wireless channel to recover the structure of the activated feature maps at the receiver. Our experiments demonstrate that the proposed method learns an effective rate control strategy that automatically adjusts the rate according to the feature maps and their entropy, as well as the channel conditions. Our method, which benefits from the entropy-aware feature map selection and the feature map pruning, provides a promising experimental result for adaptive rate control in deep JSCC. Moreover, it outperforms existing related studies in terms of image reconstruction performance.

REFERENCES

- [1] T. M. Cover, *Elements of information theory*. John Wiley & Sons, 1999.
- [2] E. Bourtsoulatze, D. B. Kurka, and D. Gündüz, "Deep joint sourcechannel coding for wireless image transmission," *IEEE Transactions* on Cognitive Communications and Networking, vol. 5, no. 3, pp. 567– 579, 2019.
- [3] D. B. Kurka and D. Gündüz, "Deep joint source-channel coding of images with feedback," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 5235–5239.
- [4] Z. Zhang, Q. Yang, S. He, M. Sun, and J. Chen, "Wireless transmission of images with the assistance of multi-level semantic information," in 2022 International Symposium on Wireless Communication Systems (ISWCS). IEEE, 2022, pp. 1–6.
- [5] D. B. Kurka and D. Gündüz, "Successive refinement of images with deep joint source-channel coding," in 2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC). IEEE, 2019, pp. 1–5.
- [6] D. B. Kurka and D. Gündüz, "Bandwidth-agile image transmission with deep joint source-channel coding," *IEEE Transactions on Wireless Communications*, vol. 20, no. 12, pp. 8081–8095, 2021.
- [7] M. Yang and H.-S. Kim, "Deep joint source-channel coding for wireless image transmission with adaptive rate control," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* IEEE, 2022, pp. 5193–5197.
- [8] Q. Zhou, R. Li, Z. Zhao, Y. Xiao, and H. Zhang, "Adaptive bit rate control in semantic communication with incremental knowledge-based harg," *IEEE Open Journal of the Communications Society*, vol. 3, pp. 1076–1089, 2022.
- [9] S. Han, J. Pool, J. Tran, and W. Dally, "Learning both weights and connections for efficient neural network," Advances in neural information processing systems, vol. 28, 2015.
- [10] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.