

Deep Joint Source-Channel Coding using Advanced Convolutional Neural Networks

Po-Hsiang Huang, Jui-Chun Roy Su, and Chia-Han Lee

National Yang Ming Chiao Tung University, Taiwan

Abstract—Deep joint source-channel coding (DeepJSCC), which leverages the power of deep neural networks to perform end-to-end optimization of source and channel coding, achieves higher robustness and efficiency compared to traditional separate source and channel coding methods. In this paper, we propose a novel DeepJSCC design called enhanced ConvNeXt-based joint source-channel coding (ECJSCC), which capitalizes on the efficiency of the ConvNeXt architecture with a novel enhanced ConvNeXt design. It is shown that the proposed ECJSCC, with much lower computational complexity, significantly outperforms SwinJSCC in terms of peak signal-to-noise ratio (PSNR), multi-scale structural similarity index measure (MS-SSIM), learned perceptual image patch similarity (LPIPS), and visual quality.

I. INTRODUCTION

Traditional communication systems have long adhered to the separation principle proposed by Shannon, which optimizes the source and channel coding independently. Unlike the traditional separation approach, joint source-channel coding (JSCC) simultaneously optimizes source coding and channel coding, enhancing the efficiency and reliability of communication systems [1],[2],[3]. However, the design of JSCC is a nontrivial task. Recently, the advance of deep learning has revolutionized this area of research, with the emergence of deep joint source-channel coding (DeepJSCC). DeepJSCC, which learns end-to-end mappings for source and channel encoding/decoding, outperforms conventional methods, especially in low signal-to-noise ratio (SNR) and high compression scenarios. Bourtsoulatzé *et al.* propose a DeepJSCC scheme for image transmission and demonstrate that deep learning-based designs outperform traditional separate source coding and channel coding techniques [4]. Despite the success of the convolutional neural network (CNN)-based DeepJSCC schemes for images transmission, they have inherent limitations due to the restricted receptive fields of convolutional layers, which struggle to capture long-range dependencies in data. By integrating the Swin Transformer [5] into the DeepJSCC framework, SwinJSCC

P.-H. Huang, J.-C. R. Su, and C.-H. Lee are with the Institute of Communications Engineering, National Yang Ming Chiao Tung University, Hsinchu, Taiwan. C.-H. Lee is also with Department of Electronics and Electrical Engineering, National Yang Ming Chiao Tung University, Hsinchu, Taiwan (e-mail: chiahan@nycu.edu.tw). This work is supported in part by MediaTek and the National Science and Technology Council (NSTC), Taiwan under grants 114-2622-8-A49-010 and 112-2221-E-A49-078-MY3.

has shown superior performance compared to earlier CNN-based schemes [6].

In this paper, we propose a novel DeepJSCC design based on ConvNeXt [7], a neural network architecture that combines the efficiency of CNNs with design elements inspired by Swin Transformers. While the Swin Transformer relies heavily on computationally expensive self-attention mechanisms, ConvNeXt achieves comparable or even superior performance with reduced complexity by leveraging the strengths of CNNs. Building upon ConvNeXt, we develop an enhanced ConvNeXt (EC) block to capture more spatial information and improve training efficiency, making it better suited for wireless image transmission. The proposed *enhanced ConvNeXt joint source-channel coding (ECJSCC)* not only outperforms the state-of-the-art SwinJSCC in terms of image quality but also offers faster processing speed with a smaller model size. Simulation results demonstrate that the proposed ECJSCC significantly outperforms the state-of-the-art SwinJSCC under various compression ratios over both the additive white Gaussian noise (AWGN) and Rayleigh block fading channels by 0.5 to 5 dB peak signal-to-noise ratio (PSNR). The proposed ECJSCC also significantly outperforms SwinJSCC in terms of multi-scale structural similarity index measure (MS-SSIM), learned perceptual image patch similarity (LPIPS), and visual quality. Furthermore, the proposed ECJSCC lowers the computational complexity by 32%, reduces the number of model parameters by 54%, and improves the inference time by 59% compared to SwinJSCC.

II. SYSTEM MODEL

Let H and W respectively represent the height and width of the input image and let $N = 3 \times H \times W$. Also, denote C_o as the output channel size of the last layer in the encoder, and H_ℓ and W_ℓ respectively as the height and width of the latent representation. Let $K = \frac{C_o}{2} \times H_\ell \times W_\ell$. Note that K is typically less than N , and $\frac{K}{N}$ is defined as the channel bandwidth ratio (CBR). The joint source-channel encoder maps the real-valued input image $\mathbf{x} \in \mathbb{R}^N$ to the vector of complex-valued symbols $\mathbf{z} \in \mathbb{C}^K$ by the encoding function $f_\theta : \mathbb{R}^N \rightarrow \mathbb{C}^K$, where θ represents the parameters of the encoding function. An average transmit power constraint P , which is set to 1, is applied at the encoder output before the signal is transmitted to ensure that $\frac{1}{K} \mathbb{E}_{\mathbf{z}}[\|\mathbf{z}\|_2^2] \leq P$, where $\mathbb{E}_{\mathbf{z}}$ denotes the expected

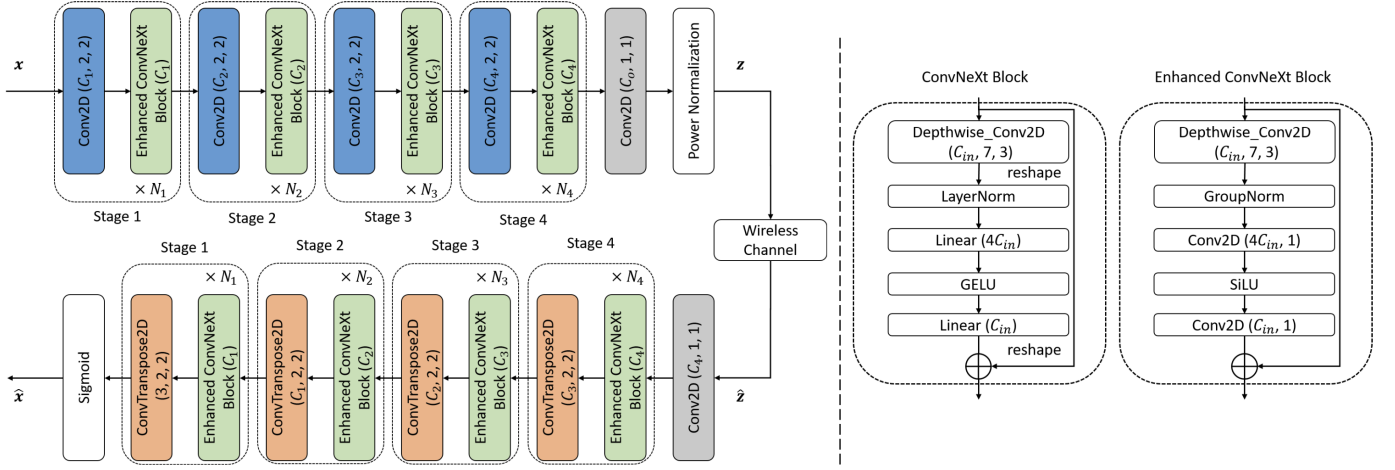


Fig. 1: The proposed ECJSCC, the original ConvNext block, and the proposed enhanced ConvNeXt block.

value over the distribution of the encoded symbols $\mathbf{z}$, and $\|\mathbf{z}\|_2$ represents the L2 norm of the vector $\mathbf{z}$. To satisfy the power constraint, $\mathbf{z} = \sqrt{KP} \frac{\tilde{\mathbf{z}}}{\sqrt{\tilde{\mathbf{z}}^* \tilde{\mathbf{z}}}}$, where $\tilde{\mathbf{z}}$ is the output of the last layer of the encoder and $\tilde{\mathbf{z}}^*$ is the conjugate transpose of $\tilde{\mathbf{z}}$. The transmitted signal $\mathbf{z}$ passes through the wireless channel modeled by the function η , and the received signal $\hat{\mathbf{z}} \in \mathbb{C}^K$ is expressed as $\eta(\mathbf{z})$. For the AWGN channel, $\eta(\mathbf{z}) = \mathbf{z} + \mathbf{n}$, where $\mathbf{n} \sim \mathcal{CN}(0, \sigma^2 \mathbf{I}_K)$ represents a noise vector of independent and identically distributed (i.i.d.) complex Gaussian noise with zero mean and variance σ^2 ; $\mathbf{I}_K$ is the identity matrix with size $K \times K$. For the Rayleigh block fading channel, the fading gain h is random but stays constant throughout the transmission block, i.e., $\eta(\mathbf{z}) = h\mathbf{z} + \mathbf{n}$, where h is sampled from the complex normal distribution with zero mean and unit variance, i.e., $h \sim \mathcal{CN}(0, 1)$. SNR is determined by the ratio of the average power of the normalized transmitted signal P to the noise power σ^2 , expressed as $\text{SNR} = 10 \log_{10} \frac{P}{\sigma^2}$ (dB). At the receiver, the joint source-channel decoder decodes the received signal $\hat{\mathbf{z}}$ to the reconstructed image $\hat{\mathbf{x}} \in \mathbb{R}^N$ by the decoding function $g_\phi: \mathbb{C}^K \rightarrow \mathbb{R}^N$, where ϕ represents the parameters of the decoding function.

The goal is to jointly optimize the encoding function f_θ and the decoding function g_ϕ to find the optimal parameter pair (θ^*, ϕ^*) . The problem is formulated to minimize the average loss between the input image $\mathbf{x}$ and the reconstructed image $\hat{\mathbf{x}}$ as $(\theta^*, \phi^*) = \arg \min_{f_\theta, g_\phi} \mathbb{E}_{p(\mathbf{x}, \hat{\mathbf{x}})} [L(\mathbf{x}, \hat{\mathbf{x}})]$, where $p(\mathbf{x}, \hat{\mathbf{x}})$ represents the joint probability of the input image $\mathbf{x}$ and the reconstructed image $\hat{\mathbf{x}}$, and $L(\mathbf{x}, \hat{\mathbf{x}})$ represents the loss function between the input image $\mathbf{x}$ and the reconstructed image $\hat{\mathbf{x}}$.

III. PROPOSED ECJSCC

The architecture of the proposed ECJSCC is illustrated in Fig. 1. At the transmitter, the pre-processed input image $\mathbf{x} \in \mathbb{R}^N$ undergoes an L -stage encoding process. Each

encoding stage progressively reduces the feature map size while extracting meaningful features. In stage i , a Conv2D layer and N_i enhanced ConvNeXt blocks are employed. The Conv2D layer having C_i output channels, kernel size 2×2 , and stride 2 is applied for downsampling to reduce both the height and the width to half of their original dimension while increasing the channel size. The enhanced ConvNeXt block, with C_i input channels, is modified from the ConvNeXt block to enhance feature extraction and processing (see Fig. 1). Another Conv2D layer with the kernel size of 1 and the stride of 1 is used in the final layer of the encoder to further compress the data to achieve the desired output channels C_o while maintaining the feature map size. Then the real-valued latent representation, with dimension $2K$, is reshaped to the complex-valued vector. After the power normalization, the complex-valued vector $\mathbf{z} \in \mathbb{C}^K$ is transmitted. At the receiver, the received signal $\hat{\mathbf{z}} \in \mathbb{C}^K$ is reshaped back to the real-valued latent representation. A Conv2D operation is utilized to expand the channels to match the output channels from the last stage of the encoder. Subsequently, an L -stage decoding process is performed. Decoding stage i consists of N_i enhanced ConvNeXt blocks and a 2D-transposed convolution (ConvTranspose2D) layer for upsampling. Finally, the sigmoid activation function is employed to map the reconstructed image values to the range $[0, 1]$.

Now let us look at how the proposed enhanced ConvNeXt block is different from the conventional ConvNeXt block. The comparison is shown in Fig. 1. The depthwise Conv2D layers are configured with C_i output channels, kernel size 7×7 , and padding size 3. Firstly, to enhance performance and efficiency, the linear layers are replaced with 1×1 Conv2D layers. The reason is that the linear layer flattens the feature maps such that the inherent structure and the spatial relationship presented in the image are lost. On the contrary, the 1×1 convolution applies a filter to each pixel inde-

pendently across all channels, which effectively learns and combines information from different channels while maintaining the feature map size. Furthermore, the layer normalization (LayerNorm) is substituted with the group normalization (GroupNorm) in the enhanced ConvNeXt block. LayerNorm, commonly used in architectures like recurrent neural networks (RNNs) and transformers, normalizes inputs across features for each training example independently. This helps stabilize the network training by reducing the internal covariate shift. For text data which is processed sequentially, LayerNorm ensures consistent feature scaling across layers. However, LayerNorm performs normalization across the entire feature map, which can be less effective for images requiring the preservation of local structures and spatial relationships. GroupNorm, on the other hand, performs normalization across groups of channels to allow normalization with finer granularity for more stable training and improved performance, especially when dealing with a small batch size. Finally, the Gaussian error linear unit (GELU) activation function is replaced with the sigmoid linear unit (SiLU) activation function. Although GELU is popular due to its capability to introduce non-linearity smoothly, SiLU has been demonstrated to offer superior performance in various deep learning tasks. By combining the advantages of abovementioned substituted functions, the proposed enhanced ConvNeXt block enables more effective feature extraction while accurately preserving the spatial hierarchy, offering superior performance compared to the original ConvNeXt block for image processing tasks.

The loss function is designed to balance the image quality and the robustness against channel variations, ensuring that the model can effectively compress and transmit images while maintaining high fidelity. It is suggested that using the L1 loss can lead to a better performance than using the mean squared error (MSE) loss in image restoration tasks [8]. Here we adopt the L1 Charbonnier loss function defined as $L(\mathbf{x}, \hat{\mathbf{x}}) = \sqrt{(\mathbf{x} - \hat{\mathbf{x}})^2 + \epsilon}$, where ϵ is a small value (10^{-6}) to ensure that the Charbonnier penalty is never zero. Using the L1 Charbonnier loss function not only preserves image details like the L1 loss but also offers a smoother function, making the model more robust to noise and outliers during training [9].

IV. PERFORMANCE EVALUATIONS

A. Simulation Setup

The CelebA-HQ dataset [10] is adopted for performance evaluations. The dataset consists of 30,000 high-resolution celebrity face images, and is divided into 24,000 images for training, 3,000 images for validation, and 3,000 images for testing. The images in the dataset are resized to 256×256 pixels. In the simulations, C_o is set to 48 and 24, which respectively correspond to CBR values of 1/32 and 1/64. In the proposed ECJSCC, there are four stages in both the encoding and decoding processes with $[N_1, N_2, N_3, N_4] = [2, 2, 4, 2]$ and $[C_1, C_2, C_3, C_4] = [128, 192, 256, 320]$.

The proposed ECJSCC is trained with SNR uniformly distributed between -5 dB and 20 dB. All neural network models are trained with a batch size of 8 and the Adam optimizer with a learning rate of 10^{-4} , $\beta_1 = 0.9$, and $\beta_2 = 0.999$. The training epoch is set to 100. All the simulations, including the training and the testing, are performed on a personal desktop computer equipped with Intel Core i7-11700 @ 2.50 GHz, 32G DRAM, and NVIDIA GeForce RTX 3060 graphics processing unit (GPU).

The following metrics are used for performance evaluations: PSNR, MS-SSIM [11], and LPIPS [12]. SwinJSCC with SNR adaptation (labeled as “SwinJSCC w/ SA”) is used for comparison. Note that in this paper, whenever we mention SwinJSCC, we mean SwinJSCC w/ SA.

B. Performance over the AWGN Channel

Fig. 2 and Fig. 3 compare the proposed ECJSCC with SwinJSCC under the AWGN channel when the CBR are respectively 1/32 and 1/64. Fig. 2(a) shows that the proposed ECJSCC consistently outperforms SwinJSCC in terms of PSNR. This trend continues as the SNR increases, with the proposed ECJSCC reaching a PSNR of about 34 dB at 20 dB SNR, compared to about 32.5 dB for SwinJSCC. Similarly, the MS-SSIM results shown in Fig. 2(b) further highlight that the proposed ECJSCC outperforms SwinJSCC. Fig. 2(c) shows that, at lower SNR, the proposed ECJSCC achieves lower LPIPS scores compared to SwinJSCC, indicating a better image quality preservation in low SNR regions. As the SNR increases, the proposed ECJSCC continues to maintain lower LPIPS scores than SwinJSCC, showing its robustness in improving the image quality across different noise levels.

From Fig. 3, we observe a trend similar to that in Fig. 2. However, with an increased compression ratio, the performance gap between the proposed ECJSCC and SwinJSCC gets larger. Based on the simulation results, it can be concluded that compared to SwinJSCC, the proposed ECJSCC is much more effective in transmitting images with high compression ratio.

C. Performance over the Rayleigh Block Fading Channel

Fig. 4 illustrates the performance comparison of the proposed ECJSCC with SwinJSCC over the Rayleigh block fading channel. Fig. 4(a) shows the significant advantage of the proposed ECJSCC, maintaining a high PSNR of 21.5 dB even when the SNR is as low as -5 dB. In contrast, SwinJSCC struggles at low SNR, with PSNR only about 20 dB. As the SNR improves, the performance gap between the proposed ECJSCC and SwinJSCC further enlarges, with the proposed ECJSCC peaking 31.5 dB PSNR at 20 dB SNR while SwinJSCC reaches just about 26.5 dB PSNR. In addition, Fig. 4(b) demonstrates that the proposed ECJSCC outperforms SwinJSCC in terms of MS-SSIM. Furthermore, as shown in Fig. 4(c), the proposed ECJSCC once again outperforms SwinJSCC in low SNR, achieving an LPIPS score around 0.31 at -5 dB SNR, compared to approximately 0.39 for SwinJSCC.

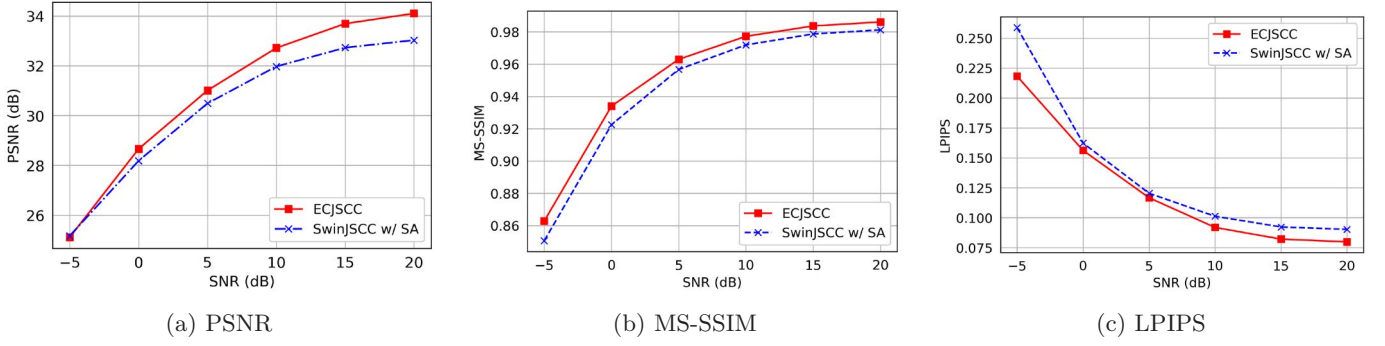


Fig. 2: Performance comparison of the proposed ECJSCC with SwinJSCC over the AWGN channel. CBR=1/32.

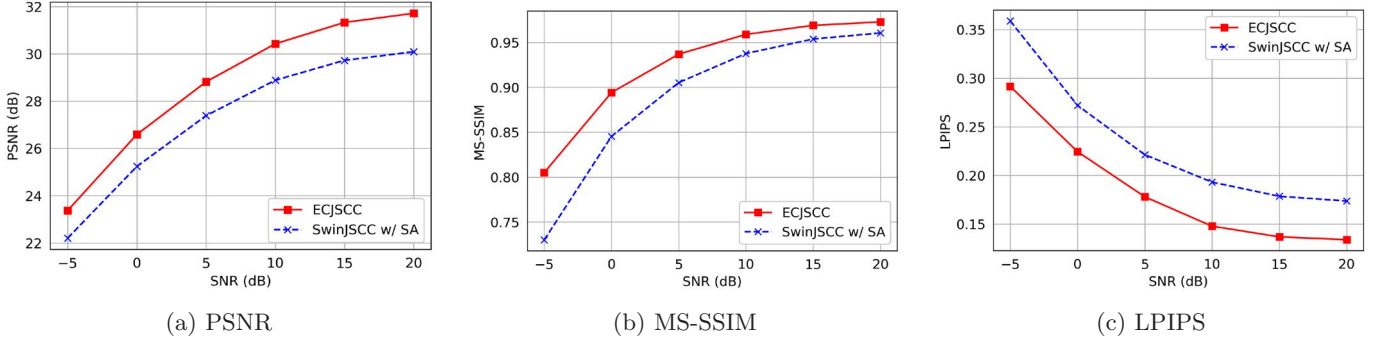


Fig. 3: Performance comparison of the proposed ECJSCC with SwinJSCC over the AWGN channel. CBR=1/64.

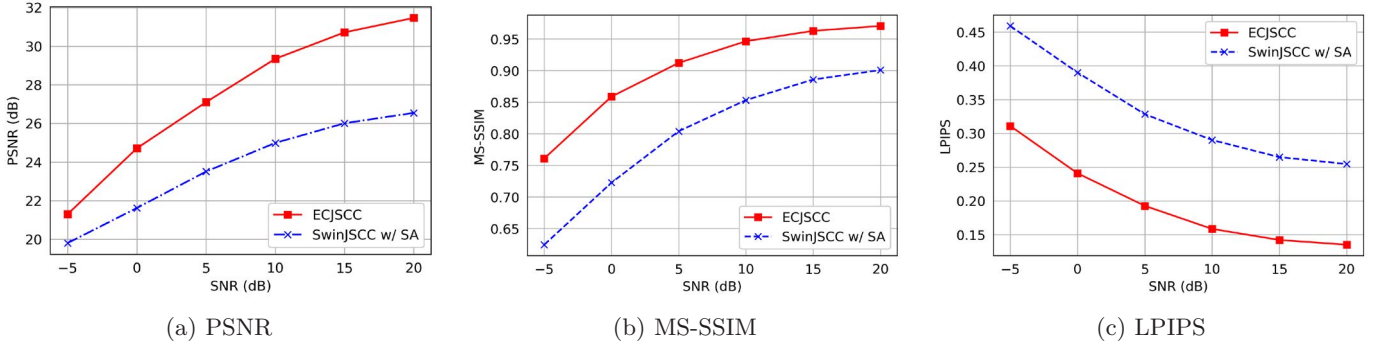


Fig. 4: Performance comparison of the proposed ECJSCC with SwinJSCC over the Rayleigh block fading channel. CBR=1/32.

D. Examples of Reconstructed Images

Fig. 5 shows examples of the reconstructed images over the AWGN channel when the CBR is 1/32. When the SNR is 15 dB, both SwinJSCC and ECJSCC perform quite well. However, when the SNR becomes -5 dB, the images reconstructed by SwinJSCC are blurred, with less distinct facial features. In contrast, the proposed ECJSCC delivers sharper images with better preservation of details.

E. Computational Complexity

To evaluate the computational complexity of the proposed ECJSCC, we use the multiply-accumulate operations (MACs) as the metric. As shown in TABLE I,

TABLE I: Computational complexity. CBR = 1/32.

Scheme	MACs	# Parameters	Inference time
Proposed ECJSCC	22.81×10^9	12.88×10^6	14.51 ms
SwinJSCC w/ SA	33.60×10^9	28.17×10^6	35.31 ms

compared to SwinJSCC w/ SA, the proposed ECJSCC reduces the computational complexity by 32%, decreases the number of model parameters by 54%, and improves the inference time by an impressive 59%.

V. CONCLUSIONS

In this paper, we have proposed a novel DeepJSCC design called ECJSCC. The proposed ECJSCC surpasses the state-of-the-art SwinJSCC not only in performance



Fig. 5: Comparison of reconstructed images of the proposed ECJSCC with SwinJSCC over the AWGN channel. CelebA-HQ dataset. CBR = 1/32.

metrics such as PSNR, MS-SSIM, and LPIPS but also in reduced complexity and shorter inference time, making it more suitable for real-time communication scenarios.

REFERENCES

- [1] A. Goldsmith, "Joint source/channel coding for wireless channels," in *Proc. IEEE 45th Vehicular Technology Conference*, July 1995, pp. 614–618.
- [2] M. Fresia, F. Pérez-Cruz, H. V. Poor, and S. Verdú, "Joint source and channel coding," *IEEE Signal Processing Magazine*, vol. 27, no. 6, pp. 104–113, Nov. 2010.
- [3] L. Deng, Z. Shi, O. Li, and J. Ji, "Joint coding and adaptive image transmission scheme based on DP-LDPC codes for IoT scenarios," *IEEE Access*, vol. 7, pp. 18 437–18 449, Jan. 2019.
- [4] E. Bourtsoulatzé, D. Burth Kurka, and D. Gündüz, "Deep joint source-channel coding for wireless image transmission," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 3, pp. 567–579, May 2019.
- [5] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin Transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2021, pp. 9992–10 002.
- [6] K. Yang, S. Wang, J. Dai, X. Qin, K. Niu, and P. Zhang, "SwinJSCC: Taming swin transformer for deep joint source-channel coding," *IEEE Transactions on Cognitive Communications and Networking*, vol. 11, no. 1, pp. 90–104, Feb. 2025.
- [7] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 11 966–11 976.
- [8] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *IEEE Transactions on Computational Imaging*, vol. 3, no. 1, pp. 47–57, Mar. 2017.
- [9] H. Yuan, W. Xu, Y. Wang, and X. Wang, "SNR-independent joint source-channel coding for wireless image transmission," 2023. [Online]. Available: <https://arxiv.org/abs/2306.15183>
- [10] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," in *Proc. International Conference on Learning Representations (ICLR)*, 2018.
- [11] Z. Wang, E. Simoncelli, and A. Bovik, "Multi-scale structural similarity for image quality assessment," in *Proc. The Thirty-Seventh Asilomar Conference on Signals, Systems & Computers*, vol. 2, Nov. 2003, pp. 1398–1402.
- [12] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018, pp. 586–595.