

# Latent Feature-Guided Conditional Diffusion for Generative Image Semantic Communication

Zehao Chen\*, Xinfeng Wei\*, Haonan Tong\*, Zhaohui Yang†, and Changchuan Yin\*

\* Beijing Laboratory of Advanced Information Network, Beijing University of Posts and Telecommunications, Beijing, China

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

Emails: {chenzhzhz, xinfengwei, hntong, and ccyin}@bupt.edu.cn, yang\_zhaohui@zju.edu.cn

**Abstract**—Semantic communication is proposed and expected to improve the efficiency of massive data transmission over sixth generation (6G) networks. However, existing image semantic communication schemes are primarily focused on optimizing pixel-level metrics, while neglecting the crucial aspect of region of interest (ROI) preservation. To address this issue, we propose an ROI-aware latent representation-oriented image semantic communication (LRISC) system. In particular, we first map the source image to latent features in a high-dimensional semantic space, these latent features are then fused with ROI mask through a feature-weighting mechanism. Subsequently, these features are encoded using a joint source and channel coding (JSCC) scheme with adaptive rate for efficient transmission over a wireless channel. At the receiver, a conditional diffusion model is developed by using the received latent features as conditional guidance to steer the reverse diffusion process, progressively reconstructing high-fidelity images while preserving semantic consistency. Moreover, we introduce a channel signal-to-noise ratio (SNR) adaptation mechanism, allowing one model to work across various channel states. Experiments show that the proposed method significantly outperforms existing methods, in terms of learned perceptual image patch similarity (LPIPS) and robustness against channel noise, with an average LPIPS reduction of 43.3% compared to DeepJSCC, while guaranteeing the semantic consistency.

**Index Terms**—Image semantic communication, joint source-channel coding, conditional diffusion model, semantic consistency, adaptive coding.

## I. INTRODUCTION

With the development of sixth generation (6G) networks, the demands for massive connectivity and low-latency data transmission are rapidly rising [1]. Traditional bit stream-based methods struggle with massive data and the increasing number of device connections [2]–[4]. To address this issue, semantic communication, which focuses on conveying the underlying meaning rather than raw data, has emerged as a promising solution to reduce transmission redundancy and adapt to dynamic network environments. Generative models, combined with joint source-channel coding (JSCC), improve transmission efficiency and robustness with enabling semantic extraction and coding process. By effectively capturing the characteristics of data distribution, generative models can significantly mitigate the semantic distortion in deep joint source-channel coding (DeepJSCC) [5], demonstrating breakthrough potential in the field of generative semantic communication [5]–[7]. The works in [7] first achieved semantic compression by refining images by key semantic features, using semantic features as

guiding signals during the decoding process. Furthermore, to enhance the perceptual similarity of reconstruction, the study in [8] extracted multimodal features and combined them with channel state information as conditions to guide the diffusion model in gradually transforming initial random noise into the final reconstructed data. To realize rate adaptive control, [9] proposed a generative DeepJSCC approach, utilizing entropy models to estimate entropy of transmitted symbols, and use these symbols to guide the diffusion process for image reconstruction. [10] proposed a novel generative communication system that leverages diffusion models and stochastic posterior sampling to enhance perceptual quality and robustness. Despite the improvements achieved by methods in [5]–[8], [11], they typically pursue average optimality in overall reconstruction quality, neglecting certain regions that are particularly critical for tasks or human perception—known as Regions of Interest (ROI). These regions carry the most semantically valuable information in an image, and accurately restoring the semantic content of ROI is more important than reconstructing fine details across the entire image. Moreover, these methods typically train separate models for discrete SNR levels, lacking explicit mechanisms to adapt to varying channel conditions - a critical limitation for practical deployment.

Inspired by [9], we propose a generative semantic communication system that incorporates adaptive ROI enhancement for image reconstruction. The key contributions of this work are: (1) We propose a ROI-aware generative image semantic communication framework. The method maps the original image into a latent feature, thereby effectively reducing the amount of data transmitted. Additionally, an ROI importance weighting mechanism is introduced to enhance specific regions of the latent representation, enabling prioritized representation and reconstruction of key semantic areas. (2) To improve robustness under varying channel states, we introduce an SNR-adaptive module that dynamically adjusts the deep joint source-channel coding (JSCC) encoder-decoder’s intermediate features based on real-time channel state information (CSI). This enables a single model to adapt seamlessly across different signal-to-noise ratio (SNR) conditions without requiring multiple trained instances. (3) We design a regionally optimized loss function that employs distinct optimization strategies for different image areas, mean squared error (MSE) loss is applied to key regions to preserve structural integrity and a combined MSE and

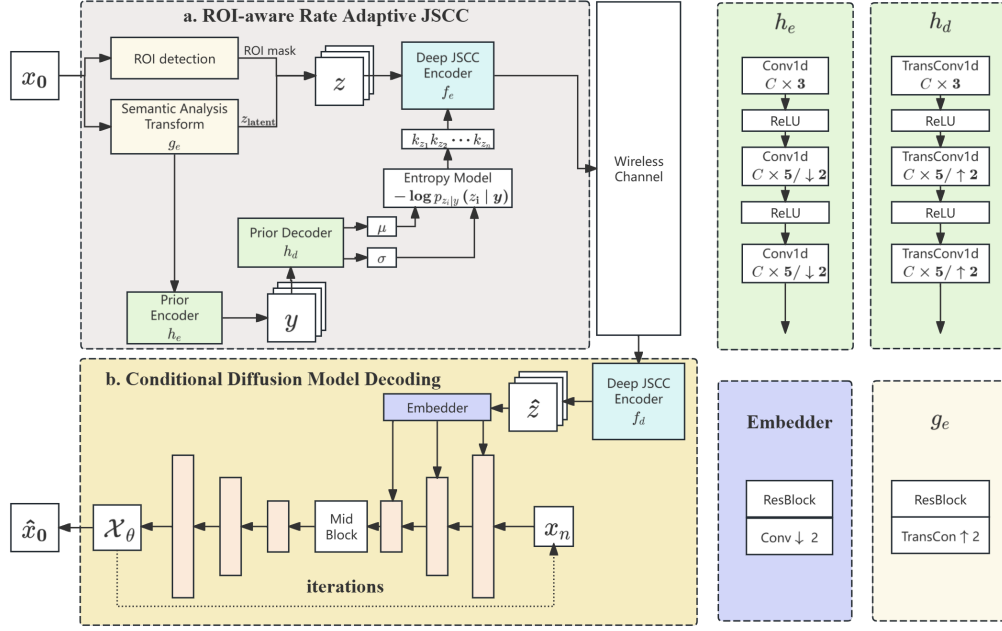


Fig. 1: The framework of semantic communication for networks.

perceptual loss enhances visual quality in non-critical regions, balancing detail fidelity and natural appearance. Simulation results demonstrate that the proposed method outperforms existing approaches under limited bandwidth scenarios, achieving better semantic consistency and perceptual quality.

The rest of this paper is organized as follows. The system model is introduced in Section II. The model architecture and training strategy is detailed in Section III. Section IV presents the simulation results and analysis to demonstrate the effectiveness of proposed system. Finally, Section V concludes the paper.

## II. SYSTEM MODEL

LRISC system integrates an ROI-aware rate-adaptive JSCC with a conditional diffusion model [9]. The latent feature  $z$  is first obtained through adaptive weighting between the ROI mask and input latent feature  $z_{\text{latent}}$ , which is then transmitted via the rate-adaptive JSCC that dynamically optimizes bandwidth allocation [12]. Finally, the received  $z$  serves as conditional guidance for the diffusion model to generate high-fidelity reconstructions [13]. This unified architecture achieves enhanced reconstruction quality through coordinated ROI-aware latent feature encoding, rate-adaptive channel transmission, and semantic-guided diffusion model decoding.

### A. ROI-aware latent feature encoding

As shown in Fig. 1, at the transmitter, a nonlinear analytical transform  $g_e$  is employed to extract the semantic feature in the latent-space from the source image  $x \in \mathbb{R}^{n \times n}$ , denoted as  $z_{\text{latent}} = g_e(x; \theta_g)$ . To optimize visual quality, we allocate more bits to the ROI, thereby enhancing the reconstruction accuracy of critical features. To distinguish between the focused area and

the background, we utilize a saliency detection network  $S$  [14] to generate an ROI mask  $m_{\text{ROI}}$  offline. A convolutional layer is then applied to smooth the saliency map, mitigating sharp boundaries and resulting in:

$$m_{\text{ROI}} = \sigma(\text{Conv}(S(x))), \quad (1)$$

where  $\sigma$  denotes the sigmoid function.

Since the spatial characteristics of the image are preserved when mapped into latent features by the encoder, we derive an ROI mask  $m_{\text{latent}}$  for  $z_{\text{latent}}$  via average pooling:

$$m_{\text{latent}} = \text{AvgPool}(m_{\text{ROI}}). \quad (2)$$

By applying the weighted mask  $m_{\text{latent}}$ , the ROI features in the latent space are amplified, thereby allocating more bits to the ROI in the resulting code stream. To further control the rate allocation, we introduce a scaling factor  $\gamma$ , which adjusts the relative importance of the ROI:

$$z = \frac{m_{\text{latent}} + \gamma}{\gamma} \otimes z_{\text{latent}}, \quad (3)$$

Here, a smaller  $\gamma$  means more bits are allocated to the ROI area in latents. What's more, to prevent the background texture from degrading, we preserve a subset of channels (e.g., the first 64 out of 256) without weighting, ensuring sufficient information is retained for background reconstruction:

$$z = z_{\text{latent ch0-ch63}} \parallel z_{\text{ch64-ch255}}. \quad (4)$$

The latent feature  $z$  consists of 256 channels, the first 64 channels are protected and the remaining channels are adaptively weighted.

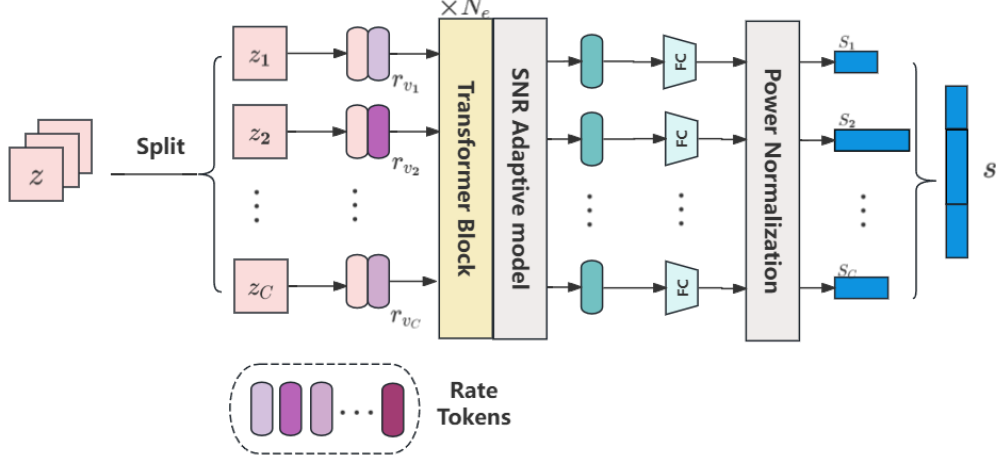


Fig. 2: The architecture of JSCC encoder.

### B. Rate-adaptive Channel Transmission

Adaptive rate allocation is achieved by leveraging entropy measurements. To enable precise entropy estimation, a hyperprior entropy model is introduced to refine entropy estimation accuracy. The hyperprior encoder  $h_e$  extracts side information  $\mathbf{y} = h_e(\mathbf{z}; \theta_h)$  capturing spatial dependencies in  $\mathbf{z}$ .  $\mathbf{z}$  contains  $k$  embedding vectors  $\mathbf{z}_i$  of dimension  $C$ . The conditional probability of  $\mathbf{z}$  given hyperprior  $\mathbf{y}$  is formally expressed as:

$$p_{\mathbf{z}|\mathbf{y}}(\mathbf{z} | \mathbf{y}) = \prod_i \left( \mathcal{N}(\mathbf{z}_i | \mu_i, \sigma_i) * \mathcal{U}\left(-\frac{1}{2}, \frac{1}{2}\right) \right) (\mathbf{z}_i), \quad (5)$$

with  $(\mu, \sigma) = h_d(\mathbf{y}; \phi_h)$ ,

where the hyperprior decoder  $h_d(\mathbf{y}; \phi_h)$  generates parameters  $(\mu, \sigma)$  that characterize a convolved Gaussian-uniform distribution for each latent element  $\mathbf{z}_i$ . This formulation accounts for quantization effects through the uniform noise term  $\mathcal{U}(-\frac{1}{2}, \frac{1}{2})$ , while the Gaussian component  $\mathcal{N}(\mathbf{z}_i | \mu_i, \sigma_i)$  enables entropy-adaptive coding.

The channel bandwidth cost  $K_z$  for transmitting  $\mathbf{z}$  can be given by

$$K_z = \sum_{i=1}^k \bar{k}_{\mathbf{z}_i} = \sum_{i=1}^k Q(k_{\mathbf{z}_i}) = \sum_{i=1}^k Q(-\beta \log p_{\mathbf{z}_i|\mathbf{y}}(\mathbf{z}_i | \mathbf{y})), \quad (6)$$

where  $\beta$  is a hyperparameter to balance channel bandwidth cost from the estimated entropy,  $\bar{k}_{\mathbf{z}_i}$  is the bandwidth consumption of  $\mathbf{z}_i$ ,  $Q$  denotes a  $2^n$ -level scalar quantization with the quantized value set as  $\mathcal{V} = \{v_1, v_2, \dots, v_{2^n}\}$ . The process by which JSCC maps the latent features  $\mathbf{z}$  into channel transmission symbol  $\mathbf{s} \in \mathbb{C}^k$  is given by

$$\mathbf{s} = f_e(\mathbf{z}, p_{\mathbf{z}_i|\mathbf{y}}(\mathbf{z}_i | \mathbf{y}); \theta_f). \quad (7)$$

To meet the energy constraints of practical communication systems, the channel transmission symbols must satisfy an average power constraint before transmission, given by  $\frac{1}{k} \sum_{i=1}^k |\mathbf{s}_i|^2 \leq P$  where  $P$  is the maximal transmission power. Given the channel transmission function  $W$ , the received symbol sequence is  $\hat{\mathbf{s}} = W(\mathbf{s})$ .

### C. Semantic-guided Diffusion Model Decoding

At the receiver, we adopt a conditional diffusion model for image reconstruction, received latent feature  $\hat{\mathbf{z}} = f_d(\hat{\mathbf{s}}; \phi_f)$  serves as the condition for the reverse process. The diffusion process (denoted by  $q$ ) gradually adds Gaussian noise to an image  $\mathbf{x}_0$  over a series of time steps, resulting in a sequence of noisy data  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ . The reverse diffusion process (denoted by  $q_\theta$ ) learns to denoise the data step by step. At step  $n$ , the two Markov processes  $q$  and  $p_\theta$  can be respectively described as

$$q(\mathbf{x}_n | \mathbf{x}_{n-1}) = \mathcal{N}(\mathbf{x}_n | \sqrt{1 - \beta_n} \mathbf{x}_{n-1}, \beta_n \mathbf{I}), \quad (8)$$

$$p_\theta(\mathbf{x}_{n-1} | \mathbf{x}_n, \hat{\mathbf{z}}) = \mathcal{N}(\mathbf{x}_{n-1} | M_\theta(\mathbf{x}_n, \hat{\mathbf{z}}, n), \beta_n \mathbf{I}). \quad (9)$$

where variable  $\beta_n$  serves as a constant hyperparameter. The reverse process is parameterized by a neural network (NN)  $M_\theta(\mathbf{x}_n, \hat{\mathbf{z}}, n)$ . We use a pixel-space prediction NN  $\mathcal{X}_\theta$  to learn to directly reconstruct image  $\mathbf{x}_0$  instead of noise  $\epsilon$  [15]. The loss function can be described as follows

$$\mathcal{L}(\theta, \mathbf{x}_0) = \mathbb{E}_{\mathbf{x}_0, n, \epsilon} \frac{\alpha_n}{1 - \alpha_n} \left\| \mathbf{x}_0 - \mathcal{X}_\theta\left(\mathbf{x}_n, \hat{\mathbf{z}}, \frac{n}{N_{\text{train}}}\right) \right\|^2, \quad (10)$$

where  $n \sim \text{Unif}\{1, \dots, N\}$ , with  $\text{Unif}$  denoting a uniform distribution over the set  $1, 2, \dots, N$ ,  $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$  and  $\alpha_n = \prod_{i=1}^n (1 - \beta_i)$ . We condition the model on the pseudo-continuous variable  $\frac{n}{N_{\text{train}}}$  which offers additional flexibility in choosing the number of denoising steps for decoding.

Once the model is trained, image can be generated by the denoising diffusion implicit model (DDIM) [16] that follows a deterministic generation, which is

$$\mathbf{x}_{n-1} = \sqrt{\alpha_{n-1}} \mathcal{X}_\theta\left(\mathbf{x}_n, \hat{\mathbf{z}}, \frac{n}{N}\right) + \sqrt{1 - \alpha_{n-1}} \epsilon_\theta\left(\mathbf{x}_n, \hat{\mathbf{z}}, \frac{n}{N}\right). \quad (11)$$

## III. MODEL ARCHITECTURE AND TRAINING STRATEGY

This section presents three parts: the NN architecture of LRISC, the system's loss function, and the multi-stage training algorithm.

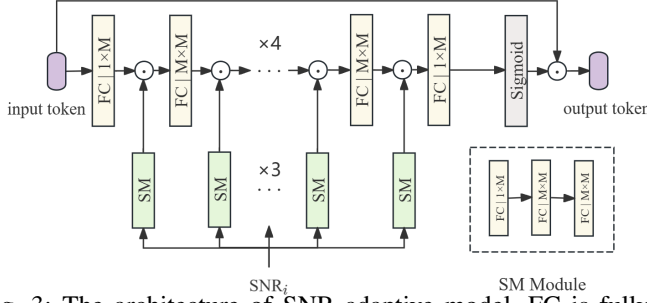


Fig. 3: The architecture of SNR adaptive model. FC is fully-connected network, “ $\odot$ ” is the element-wise product.

#### A. Model Architecture

- 1) **Prior Codec**  $h_e, h_d$ : The structure of  $h_e$  and  $h_d$  is shown in Fig. 1. It consists of fully convolutional layers followed by the ReLU activation functions.
- 2) **JSCC Codec**  $f_e, f_d$ : We adopt a pair consisting of a transformer-like JSCC encoder and decoder as  $f_e, f_d$ . The encoder consists of  $N_e$  transformer blocks and fully connected (FC) layers to implement rate allocation, as shown in Fig. 2. Specifically,  $z$  is firstly separated into patch embedding sequence  $z_1, z_2, \dots, z_c$ . Guided by the entropy model  $-\log P_{z_i|y}(z_i | y)$ , a set of learnable rate token embeddings with the same dimension as  $z_i$  are developed [12], each of which corresponds to a value in  $\mathcal{V} = \{v_1, v_2, \dots, v_{2^n}\}$ , each  $z_i$  is merged with its corresponding rate token  $r_{vi}$ . FC layers with output dimensions of  $v_q, q = 1, 2, \dots, 2^n$  are employed to map the embeddings into  $s_i$  with given dimensions.
- 3) **SNR Adaptive Model**: The SNR adaptive model is inserted to the last layer of JSCC encoder and the first layer of JSCC decoder, as shown in Fig. 3. It consists of 8 FC layers alternating with 7 SNR modulation (SM) modules [17]. SM module is a three-layered FC network, which takes the SNR for received  $z_i$  (denoted by  $\text{SNR}_i$ ) as input and transforms it into an  $c$ -dimensional vector  $sm_i$ . Multiple SM modules are cascaded sequentially in a coarse-to-fine manner, the previous modulated features are fed into subsequent SM modules.
- 4) **Conditional Diffusion Model**: The denoising module utilizes a U-Net structure [18], [19]. Each U-Net unit is composed of two ResNet blocks, an attention block, and a convolutional upsampling and downsampling block. The conditional diffusion model network employs six U-Net units in both the downsampling and upsampling paths. During the downsampling phase, the channel dimension expands according to the formula  $64 \times j$ , where  $j$  denotes the layer index ( $j = 1, 2, \dots, 6$ ). The upsampling units follow the reverse order. For embedding condition  $z$ , we employ ResNet blocks and transpose convolutions to upscale  $z$  to match the inputs' dimensions of the initial four U-Net downsampling units.

#### B. Loss Function

The optimization objective is to achieve a trade-off between compression performance and image reconstruction quality. To enhance the focus on ROI area while maintaining the perceptual quality of the overall image, a regionally differentiated distortion function is designed under the guidance of an ROI mask to optimize the reconstruction quality of both ROI area and the background area. Specifically, separate distortion losses were defined for the ROI region  $d_{\text{ROI}}$  and the background region  $d_{\text{BG}}$ . The ROI region emphasizes pixel-level reconstruction accuracy in critical semantic areas, as shown in below:

$$d_{\text{ROI}}(x, \hat{x}) = m_{2D} \otimes \text{MSE}(x, \hat{x}), \quad (12)$$

MSE distortion measures the pixel-level differences between a reconstructed image and the original image, ensuring image fidelity. For background regions, a combination of MSE and perceptual loss terms is adopted to enhance the overall perceptual quality of the image, ensuring that the reconstructed image aligns more closely with human visual perception in terms of structure and texture, resulting in:

$$d_{\text{BG}}(x, \hat{x}) = \text{MSE}(x, \hat{x}) + \eta \cdot d_P(x, \hat{x}), \quad (13)$$

Here,  $d_P(\cdot, \cdot)$  represents the perceptual loss term, which employs LPIPS as the perceptual metric. It is used to balance the trade-off between MSE distortion and perceptual loss. The overall distortion  $d(x_0, \hat{x}_0)$  of the reconstructed image can be expressed as:

$$\begin{aligned} L &= D + \lambda \cdot R \\ &= \mathbb{E}_{x_0} [-\lambda \log p_{z|y}(z | y) - \lambda \log p_y(y) \\ &\quad + d(x_0, \hat{x}_0) + d(x_0, \bar{x}_0)], \end{aligned} \quad (14)$$

Here,  $\lambda$  denotes the hyperparameter that controls the trade-off between rate and distortion,  $R$  represents the total channel transmission rate, and  $D$  indicates the end-to-end distortion.  $d(x_0, \hat{x}_0)$  measures the distortion between the original image and the reconstructed image, while  $d(x_0, \bar{x}_0)$  measures the distortion between the original image and the compressed image.

#### C. Training Strategy

To ensure training stability and enhance overall performance, we propose a multi-stage training strategy. Initially, we train each module individually [20] to reduce complexity and facilitate convergence [21]. Once all modules have been trained separately, we fine-tune the entire model. The complete multi-stage training process is shown in Algorithm 1. First, train  $g_e, g_d, h_e$  and  $h_d$ . Secondly, train JSCC codecs  $f_e$  and  $f_d$  with channel data. At last, fine-tune all modules from the aforementioned two steps in an end-to-end optimization manner.

### IV. SIMULATION ANALYSIS AND DISCUSSIONS

#### A. Simulation Settings

For our experiments, we use the COCO dataset for training, and we select 50,000 images, which are randomly cropped to

---

**Algorithm 1** Training procedure for the proposed LRISC

---

**Input:** Training dataset  $X$ , the Lagrange multiplier  $\lambda$  on the rate term, perception and distortion trade-off parameter  $\eta$ , scaling factor  $\beta$  from entropy to channel bandwidth cost and learning rate  $l_r$ .

```
1: Stage 1: Train  $g_e, g_d, h_e, h_d$ 
2: Randomly initialize all parameters and freeze the parameters of  $f_e$  and  $f_d$ 
3: for training iteration 1 to  $N_{\text{train}}$  do
4:   sample  $\mathbf{x} \sim p_{\mathbf{X}}, n \sim \mathcal{U}(0, 1, 2, \dots, N_{\text{train}}), \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ 
5:    $\bar{\mathbf{x}}_n = \sqrt{\alpha_n} \mathbf{x}_0 + \sqrt{1 - \alpha_n} \epsilon$ 
6:    $\bar{\mathbf{x}}_0 = \mathcal{X}_\theta(\bar{\mathbf{x}}_n, n/N_{\text{train}}, \mathbf{z})$ 
7:    $\mathcal{L} = \mathbb{E}_{\mathbf{x}_0} [d(\mathbf{x}_0, \bar{\mathbf{x}}_0) + \lambda [-\log p_{\mathbf{z}|\mathbf{y}}(\mathbf{z} | \mathbf{y}) - \log p_{\mathbf{y}}(\mathbf{y})]$ 
8:   Update the parameters  $(\theta_g, \phi_g, \theta_h, \phi_h)$ 
9: end for
10: Stage 2: Train  $f_e, f_d$ 
11: Load and freeze the parameters trained in Stage 1 and randomly initialize  $f_e$  and  $f_d$ 
12: for each epoch do
13:   Sample  $\mathbf{x} \sim p_{\mathbf{X}}$ 
14:   Calculate the loss function based on Eq.(14)
15:   Update parameters:  $(\theta_f, \phi_f)$ 
16: end for
17: Stage 3: Fine-tune the whole model
18: Load the parameters trained in the previous stages
19: for each epoch do
20:   Repeat steps 14 to 16
21:   Update parameters  $(\theta_g, \theta_h, \phi_g, \phi_h, \theta_f, \phi_f)$ 
22: end for
Output: Parameters  $(\theta_g^*, \phi_g^*, \theta_h^*, \phi_h^*, \theta_f^*, \phi_f^*)$ 
```

---

256  $\times$  256, and the Kodak dataset for testing. We implement the proposed LRISC model using the PyTorch framework. We exploit the Adam optimizer with an initial learning rate of  $1 \times 10^{-4}$ , and the batch size is set to 16. In the first stage, we randomly initialize the model parameters and train  $g_e, g_d, h_e$  and  $h_d$ , the training steps set to  $10^6$ . In the second stage, we first train JSCC codec  $f_e$  and  $f_d$  with a fixed channel state (SNR = 10dB) except for the snr adaptive module. Then, the whole JSCC codec is trained with snr adaptive module with variable channel state. Finally, during fine-tuning, all model parameters are unfrozen, and the model is trained on the full COCO training dataset for another 20 epochs, with a gradually decaying learning rate. The scaling factor  $\beta$  is set to 0.2 at SNR = 10dB, the perception and distortion trade-off parameter  $\eta$  is set to 0.5. The training process is conducted on an NVIDIA RTX 4090 GPU to accelerate computation.

### B. Performance Comparison

We compare our proposed method with two widely recognized baselines: 1) the classical DeepJSCC model optimized using MSE distortion [22] and 2) the traditional source-channel separation scheme. Specifically, the source-channel separation scheme uses the BPG for source coding, combined with Low-Density Parity-Check (LDPC) for channel coding, which is marked as “BPG + LDPC”.

For the performance evaluation, we utilize two commonly employed metrics: LPIPS and peak signal-to-noise-ratio (PSNR). LPIPS is used for measuring perceptual similarity, as it correlates more closely with human visual perception

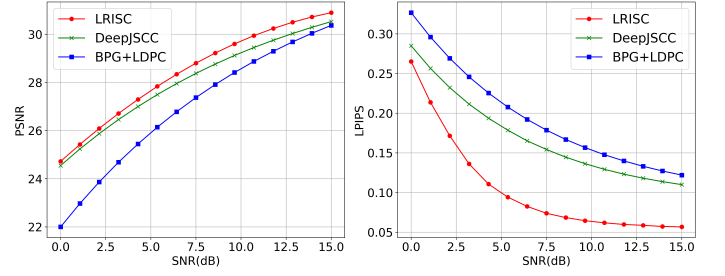


Fig. 4: PSNR and LPIPS performance versus SNR.

than traditional pixel-based metrics like PSNR. While PSNR provides a quantitative measure of the reconstruction quality based on pixel differences.

Fig. 4 compares the performance across varying SNR levels. LRISC consistently achieves the highest PSNR across the entire SNR range. Notably, at lower SNR values, LRISC and DeepJSCC demonstrate significant robustness, and achieves a reduction of nearly 43.3% in LPIPS compared to DeepJSCC. This improvement highlights the effectiveness of LRISC’s optimized JSCC strategy in preserving image quality under noisy channel conditions. When examining LPIPS, the performance gap between LRISC and the baseline methods widens as SNR increases. This improvement is because LRISC generates images’ perceptual information, with more semantics aligning to the original images.

Fig. 6 demonstrates the performance comparison for different methods in terms of PSNR as the channel bandwidth ratio (CBR) varies. Specifically, LRISC consistently outperforms the DeepJSCC and “BPG + LDPC” baselines in terms of PSNR. This improvement is attributed to the enhanced compression and adaptive coding in our LRISC, which enables higher reconstruction quality with lower CBR. Additionally, LRISC achieves the lowest LPIPS value, indicating superior perceptual fidelity compared to other methods. In contrast, the BPG+LDPC baseline exhibits the highest LPIPS values, indicating that while it maintains acceptable pixel-level quality, its perceptual quality remains suboptimal.

Fig. 5 shows the image transmission quality under low CBR conditions. Fig. 5 (a) shows the transmitted original image, while Fig. 5 (b) presents the ROI map generated by our system. The ROI map highlights the key areas of the image, which are allocated more bits to ensure reconstruction accuracy. In Fig. 5 (c), we demonstrate the transmitted image under a low channel bit rate of 0.017 and an SNR of 10. It is clear that, even under low CBR conditions, our system effectively preserves the semantic content of the image, particularly in the ROI region. Despite some minor distortions, the overall quality of the transmitted image is very close to the original. The combination of the ROI-aware mechanism and adaptive rate control enables our system to maintain high perceptual quality with low bit rate conditions, demonstrating robustness in suboptimal transmission environments while ensuring semantic consistency and visual quality.

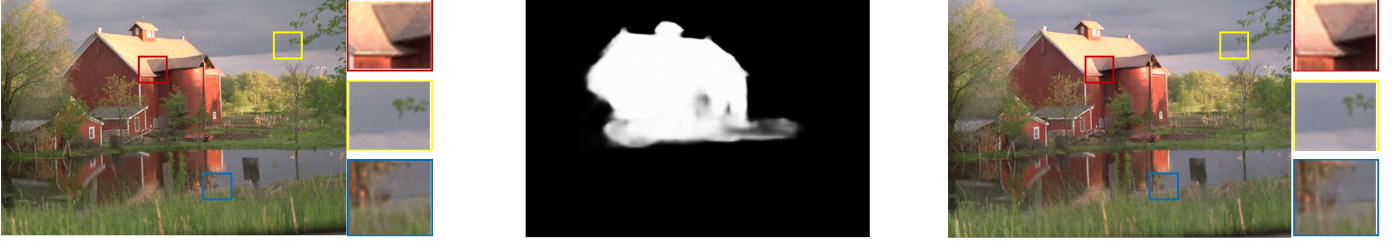


Fig. 5: (a) Original image; (b) The image's ROI area; (c) Reconstructed image with low CBR

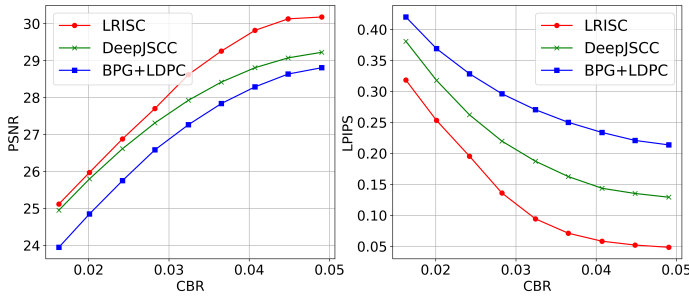


Fig. 6: PSNR and LPIPS performance versus CBR.

## V. CONCLUSION

In this paper, we propose an ROI-aware generative image semantic communication system driven by latent features LRISC. Specifically, the system incorporates an adaptive ROI-guided feature encoding mechanism that intelligently allocates transmission rates, combined with a powerful diffusion model to achieve high-fidelity image reconstruction under limited bandwidth. Simulation results demonstrate that the system not only maintains robustness in semantic image transmission but also delivers superior perceptual quality across various channel conditions.

## REFERENCES

- [1] P. Zhang, W. Xu, H. Gao, K. Niu, X. Xu, X. Qin, C. Yuan, Z. Qin, H. Zhao, J. Wei *et al.*, "Toward wisdom-evolutionary and primitive-concise 6G: A new paradigm of semantic communication networks," *Engineering*, vol. 8, pp. 60–73, 2022.
- [2] S. Barbarossa, D. Communiello, E. Grassucci, F. Pezone, S. Sardellitti, and P. Di Lorenzo, "Semantic communications based on adaptive generative models and information bottleneck," *IEEE Communications Magazine*, vol. 61, no. 11, pp. 36–41, 2023.
- [3] J. Dai, P. Zhang, K. Niu, S. Wang, Z. Si, and X. Qin, "Communication beyond transmitting bits semantics-guided source and channel coding," *IEEE Wireless Communications*, vol. 30, no. 4, pp. 170–177, 2022.
- [4] E. Grassucci, Y. Mitsufuji, P. Zhang, and D. Communiello, "Enhancing semantic communication with deep generative models: An overview," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 13 021–13 025.
- [5] E. Erdemir, T.-Y. Tung, P. L. Dragotti, and D. Gündüz, "Generative joint source-channel coding for semantic image transmission," *IEEE Journal on Selected Areas in Commun.*, vol. 41, no. 8, pp. 2645–2657, 2023.
- [6] T. Wu, Z. Chen, D. He, L. Qian, Y. Xu, M. Tao, and W. Zhang, "Cddm: Channel denoising diffusion models for wireless semantic communications," *IEEE Transactions on Wireless Communications*, vol. 23, no. 9, pp. 11 168–11 183, 2024.
- [7] L. Qiao, M. B. Mashhadi, Z. Gao, C. H. Foh, P. Xiao, and M. Bennis, "Latency-aware generative semantic communications with pre-trained diffusion models," *IEEE Wireless Communications Letters*, vol. 13, no. 10, pp. 2652–2656, 2024.
- [8] E. Lei, Y. B. Uslu, H. Hassani, and S. S. Bidokhti, "Text+ sketch: Image compression at ultra low rates," *arXiv preprint arXiv:2307.01944*, 2023.
- [9] P. Yang, G. Zhang, and Y. Cai, "Rate-adaptive generative semantic communication using conditional diffusion models," *IEEE Wireless Communications Letters*, vol. 14, no. 2, pp. 539–543, 2025.
- [10] S. Wang, J. Dai, K. Tan, X. Qin, K. Niu, and P. Zhang, "Diffcom: Channel received signal is a natural condition to guide diffusion posterior sampling," *IEEE Journal on Selected Areas in Communications*, 2025.
- [11] M. Yang, B. Liu, B. Wang, and H.-S. Kim, "Diffusion-aided joint source channel coding for high realism wireless image transmission," *arXiv preprint arXiv:2404.17736*, 2024.
- [12] J. Dai, S. Wang, K. Tan, Z. Si, X. Qin, K. Niu, and P. Zhang, "Nonlinear transform source-channel coding for semantic communications," *IEEE Journ. on Select. Areas in Commun.*, vol. 40, no. 8, pp. 2300–2316, 2022.
- [13] R. Yang and S. Mandt, "Lossy image compression with conditional diffusion models," *Advances in Neural Information Processing Systems*, vol. 36, pp. 64 971–64 995, 2023.
- [14] Z. Chen, Q. Xu, R. Cong, and Q. Huang, "Global context-aware progressive aggregation network for salient object detection," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 07, 2020, pp. 10 599–10 606.
- [15] T. Salimans and J. Ho, "Progressive distillation for fast sampling of diffusion models," *arXiv preprint arXiv:2202.00512*, 2022.
- [16] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.
- [17] 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, 2025.
- [18] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III 18*. Springer International Publishing, 2015, pp. 234–241.
- [19] Y. Zeng, X. He, X. Chen, H. Tong, Z. Yang, Y. Guo, and J. Hao, "DMCE: Diffusion model channel enhancer for multi-user semantic communication systems," in *IEEE International Conf. on Commun.*, 2024, pp. 855–860.
- [20] X. Wei, H. Tong, N. Yang, and C. Yin, "Language-oriented semantic communication for image transmission with fine-tuned diffusion model," in *2024 16th International Conference on Wireless Communications and Signal Processing (WCSP)*, 2024, pp. 1456–1461.
- [21] H. Tong, H. Li, H. Du, Z. Yang, C. Yin, and D. Niyato, "Multimodal semantic communication for generative audio-driven video conferencing," *IEEE Wireless Communications Letters*, vol. 14, no. 1, pp. 93–97, 2025.
- [22] E. Boursoulatz, D. B. Kurka, and D. Gündüz, "Deep joint source-channel coding for wireless image transmission," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 4774–4778.