JOINT SEMANTIC-CHANNEL CODING AND MODULATION FOR GENERATIVE IMAGE TRANSMISSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Digital image transmission typically optimizes source coding, channel coding, and modulation separately, limiting how source representations adapt to channel impairments. We propose SemQAM, a joint semantic-channel coding and modulation scheme for generative image transmission converting source to modulated symbols, such as standard QAM symbols. Channel impairments create two challenges: they can corrupt the latent representation at the receiver, and this corruption can compound through autoregressive coding, degrading reconstruction quality. First, SemQAM represents latent features with vector quantization (VQ), maps each VQ index to a QAM symbol, and updates the codeword corresponding to the detected symbol so that channel impairments directly affect codebook learning. Moreover, SemQAM uses the conditional expectation of received codewords to approximate the receiver reconstruction and reduce error propagation during autoregressive coding. SemQAM supports multiple SNRs and CBRs in a unified model, while also supporting conventional source coding through SemQAM-C. Experiments show that (i) under the same LDPC and QAM settings, SemQAM-C already outperforms OneDC in LPIPS and DISTS with substantially lower complexity, showing that the unified model retains competitive capability for source coding. (ii) Compared with this separated counterpart, SemQAM reduces channel uses by 18.57% at matched LPIPS and also outperforms the compared JSCC methods, demonstrating the benefit of joint coding.

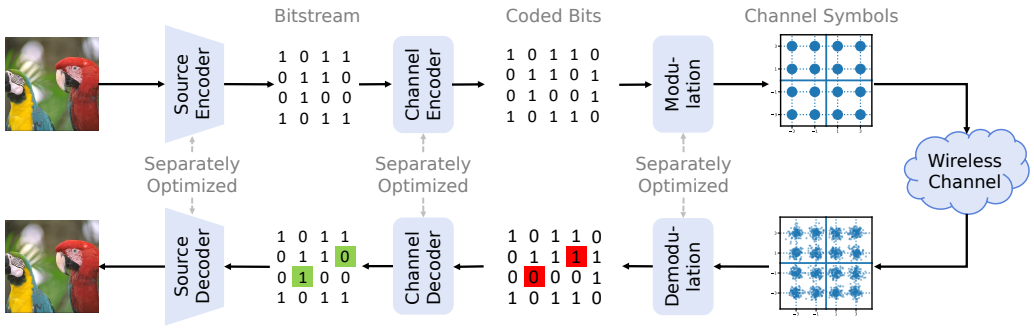
1 INTRODUCTION

Digital image transmission is an important application of modern communication systems (Gündüz et al., 2025; Xie et al., 2021). A conventional digital image transmission system consists of three separately optimized stages, as illustrated in Figure 1a. (i) Source coding (Shannon, 1959) compresses an image into as few bits as possible at the desired reconstruction quality. (ii) Channel coding (Shannon, 1948; Gallager, 1962) adds redundancy to protect these bits from transmission errors. (iii) Modulation (Hancock & Lucky, 1960) maps the protected bits to channel symbols, such as quadrature amplitude modulation (QAM) symbols, for transmission.

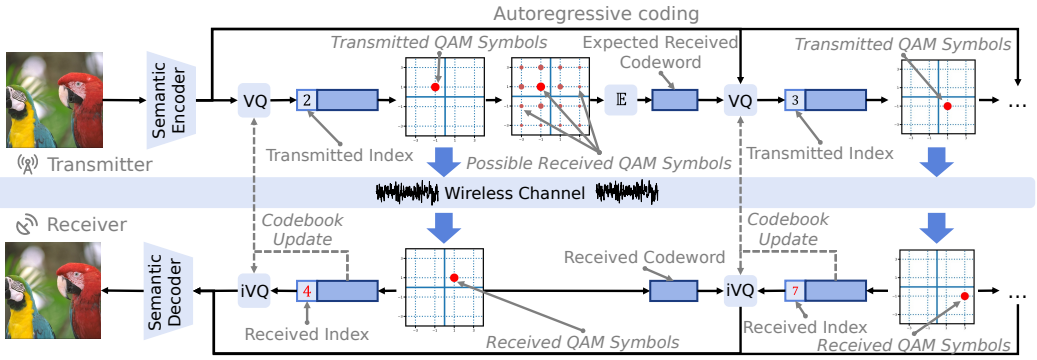
However, with limited transmission resources, source representation and error protection compete for the budget, motivating learned joint semantic-channel coding (JSCC) (Dai et al., 2022; Huang & Qin, 2026) and joint semantic-channel coding and modulation (JSCM) (Zhang et al., 2025; Ying et al., 2026) for end-to-end optimization of the transmission system. However, how QAM detection errors alter the latent representation reconstructed at the receiver and propagate through autoregressive coding remains less explored.

In this paper, we propose SemQAM, a joint **sem**antic-channel coding and modulation system for generative image transmission using **QAM**, as illustrated in Figure 1b. We observe that channel impairments degrade reconstruction quality for two reasons: they corrupt the latent representation reconstructed at the receiver, and the resulting corruption further compounds during autoregressive coding. First, SemQAM maps vector quantization (VQ) indices of image latents to QAM symbols and updates the codeword corresponding to the detected symbol. Second, SemQAM computes the expected received codeword from channel transition probabilities and uses it for subsequent autoregressive encoding. Together, these designs adapt the latent representation to channel impairments and reduce corruption accumulation during autoregressive coding, improving reconstruction quality.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) Conventional digital communication system



(b) SemQAM

Figure 1: Conventional separated transmission and SemQAM. **(a)** A conventional system separately optimizes source coding, channel coding, and modulation. **(b)** SemQAM is designed to jointly optimize end-to-end image transmission over a standard QAM constellation. SemQAM *(i)* trains VQ codewords using the detected indices at the receiver, and *(ii)* uses the expectation of received codewords to approximate the receiver reconstruction during autoregressive coding. $\mathbb{E}$, VQ, and iVQ denote conditional expectation, vector quantization, and codeword lookup, respectively.

SemQAM supports multiple signal-to-noise ratios (SNRs) and channel bandwidth ratios (CBRs) in a unified model, while also providing a noiseless source-coding mode, SemQAM-C, to accommodate diverse transmission scenarios. Experiments on Kodak at 10 dB show that (i) under the same LDPC (Gallager, 1962; Richardson et al., 2001) and QAM, OneDC (Xue et al., 2025b) requires 20.96% and 17.01% more channel uses than SemQAM-C at matched LPIPS and DISTS, respectively, while SemQAM-C uses only 5.59% of the encoder latency, 2.59% of the decoder latency, and 4.29% of the parameters, showing that the unified model preserves competitive capability for source coding. (ii) SemQAM outperforms the compared JSCC methods in LPIPS (Zhang et al., 2018), DISTS (Ding et al., 2022), and FID (Heusel et al., 2017), and reduces channel uses by 18.57% on average at matched LPIPS compared with the separated SemQAM-C system, directly demonstrating the advantage of joint coding over separate optimization.

Our contributions are summarized as follows:

- We introduce SemQAM, a joint semantic-channel coding and modulation system for generative image transmission with standard QAM, supporting multiple SNRs and CBRs as well as conventional source coding in a unified model.
- SemQAM trains VQ codewords using detected receiver indices to incorporate channel transitions into codebook learning, and uses the expectation of received codewords to approximate receiver reconstructions and reduce error propagation.
- Experiments demonstrate the advantage of joint coding over separate optimization while retaining strong performance and efficiency in conventional source coding.

2 PRELIMINARY

Digital Image Transmission. Digital image transmission aims to reconstruct the source image with high quality using as few transmitted symbols as possible. During wireless transmission, channel impairments such as noise can distort the transmitted signals and cause symbol errors, so the system must represent the image efficiently while remaining robust to these errors.

Source Coding. Source coding removes redundancy from the source representation to reduce the number of bits or symbols needed for reconstruction. Autoregressive coding is an important approach for reducing redundancy among latent features (Ballé et al., 2018; Minnen et al., 2018). In EF-LIC (Cao et al., 2026), previously reconstructed groups predict an affine transform for the current latent before VQ, removing predictable information and improving representation efficiency under a fixed VQ budget. This also makes later quantization depend on earlier reconstructions.

Channel Coding. Channel coding adds redundancy to transmitted bits so that the receiver can correct errors caused by channel impairments. Low Density Parity Check (LDPC) codes (Gallager, 1962; Richardson et al., 2001) are among the most widely used channel codes in modern communication systems (Hui et al., 2018). We therefore use LDPC to construct the separated transmission baselines in our experiments.

Modulation. Modulation converts coded bits into physical signals for transmission over the wireless channel. Quadrature amplitude modulation (QAM) (3GPP, 2020) is one of the most widely used modulation schemes and maps coded bits to a finite set of constellation points. An M-QAM constellation contains M points, so each symbol can represent $\log_2 M$ bits. For example, 1024-QAM contains 1024 constellation points and carries 10 bits per symbol, naturally matching the 1024 possible indices of a VQ codebook.

Wireless Channel. A wireless channel is the physical medium through which signals are transmitted from the transmitter to the receiver. A common baseband channel model is

$$r = hs + N,$$

where s and r denote the transmitted and received symbols, h denotes the channel response, and N denotes additive noise. The signal-to-noise ratio (SNR) is a common measure of channel conditions and is defined as

$$\mathrm{SNR} = 10 \log_{10} \frac{\mathbb{E}[|hs|^2]}{\mathbb{E}[|N|^2]} \quad \mathrm{dB}.$$

Under a fixed signal power, a larger SNR corresponds to weaker noise and a cleaner channel. For digital image transmission, the channel bandwidth ratio (CBR) measures the transmission resources relative to the source image size. For an RGB image of size $H \times W$ transmitted using n channel uses, CBR is defined as

$$\mathrm{CBR} = \frac{n}{3HW},$$

where a lower CBR indicates fewer transmission resources.

Joint Semantic-Channel Coding. Separate source and channel coding is optimal only under restrictive asymptotic conditions, motivating learned joint transmission methods for practical systems with finite resources (Shannon, 1959; Kostina & Verdú, 2013). Previous JSCC methods directly transmit continuous-valued encoded features (Bourtsoulatze et al., 2019; Dai et al., 2022; Huang & Qin, 2026). JSCM instead maps the encoded features to a finite set of digital constellation points. Under otherwise matched conditions, this constraint reduces the available signal space and can lower the achievable performance relative to continuous JSCC, while making transmission compatible with practical digital systems. As one of the most widely used digital modulation schemes, QAM has been adopted by several JSCM methods (Ying et al., 2026; Tung et al., 2022; Bo et al., 2024; Zhang et al., 2025), making QAM-based JSCM compatible with existing communication systems.

3 METHOD

3.1 SYSTEM OVERVIEW

SemQAM builds on the encoding structure of EF-LIC (Cao et al., 2026), extending it from source coding to joint semantic-channel coding and modulation. The detailed model architecture is shown in Figure 6 in Appendix B.1. The analysis transform maps an image x to the main latent $y = g_a(x)$, while the hyper analysis transform extracts the hyperlatent $z = h_a(y)$ (Ballé et al., 2018). Both latents are represented using residual vector quantization (RVQ) (van den Oord et al., 2017; Kumar et al., 2023), with the hyperlatent transmitted first to provide context for the main latent. Because channel noise can produce different reconstructions at the transmitter and receiver, the transmitter maintains an approximation $\tilde{z}$ of the receiver hyperlatent, while the receiver reconstructs $\hat{z}$ from its detected indices, producing context features $\tilde{\phi} = h_s(\tilde{z})$ and $\hat{\phi} = h_s(\hat{z})$.

The main latent y is divided into four quadtree groups $y_1, \ldots, y_4$, where later groups use earlier reconstructions as context (Cao et al., 2026). The transmitter constructs the context from its approximated receiver reconstructions $\tilde{y}_{<i}$, while the receiver uses its actual reconstructions $\hat{y}_{<i}$.

$$(\tilde{\mu}_i, \tilde{\sigma}_i) = f_i(\tilde{y}_{<i}, \tilde{\phi}), \qquad (\hat{\mu}_i, \hat{\sigma}_i) = f_i(\hat{y}_{<i}, \hat{\phi}).$$

The transmitter normalizes the current group as $u_i = (y_i - \tilde{\mu}_i) \oslash \tilde{\sigma}_i$, where $\oslash$ denotes elementwise division, and RVQ successively quantizes its residuals.

For each RVQ stage, the transmitter selects the nearest codeword to obtain an index J. Each codebook contains 1024 codewords whose indices are mapped one to one to Gray-labeled 1024-QAM symbols (Cho & Yoon, 2002), without separate channel coding between VQ and modulation. After wireless transmission, the receiver detects $\hat{J}$ and reconstructs from $c_{\hat{J}}$, while the transmitter maintains an approximation of the receiver reconstruction because $\hat{J}$ is unavailable to it. These local reconstructions are then used for later autoregressive coding, and the receiver finally reconstructs the image as $\hat{x} = g_s(\hat{y})$. Across channel conditions and transmission rates, SemQAM shares the image transforms, hyperprior, and context networks while selecting the corresponding VQ codebooks for each operating point.

3.2 VQ CODEBOOK LEARNING UNDER CHANNEL IMPAIRMENTS

In a noiseless VQ system, codeword indices are only labels (van den Oord et al., 2017). Permuting these labels does not affect reconstruction as long as the encoder and decoder use the same mapping. This symmetry is broken after the indices are mapped to QAM symbols because channel noise may change a transmitted index J into a different detected index $\hat{J}$. If a QAM detection error selects a codeword that is very different from the transmitted one, the reconstructed latent can change substantially, degrading the final image reconstruction.

SemQAM addresses this problem through receiver-side codebook learning. For one RVQ stage, let u and $\hat{u}$ denote the current residuals constructed from the transmitter and receiver reconstructions, respectively, and let c_j and s_j denote the j-th VQ codeword and its mapped QAM symbol. The receiver codebook loss is

$$\mathcal{L}_{\mathrm{cb}} = \left\| \mathrm{sg}(\hat{u}) - c_{\hat{J}} \right\|^2 . \tag{1}$$

where

$$J = \arg\min_j \|u - c_j\|^2, \quad r = hs_J + N, \quad \hat{J} = \arg\min_k |r - hs_k|^2.$$

Here, sg denotes stop-gradient. The transmitted index J determines the QAM symbol entering the channel, while the detected index $\hat{J}$ determines which VQ codeword receives the training signal. Thus, a sample assigned to one index at the transmitter can update the codeword corresponding to another index after channel transmission, incorporating the transition between transmitted and detected indices into codebook learning.

Over training samples and channel realizations, the receiver codebook learning can be written as

$$\min_{\{c_k\}} \mathbb{E}[\mathcal{L}_{\mathrm{cb}}] = \min_{\{c_k\}} \sum_j \mathbb{P}\left(J = \arg\min_\ell \|u - c_\ell\|^2 \right) \mathbb{E}\left[\left\| \mathrm{sg}(\hat{u}) - c_{\hat{J}} \right\|^2 \mid J = j \right]. \tag{2}$$

This receiver-side objective therefore incorporates the transition from transmitted index J to detected index $\hat{J}$ into codebook learning. With autoregressive coding, the detected codeword also affects the residuals and context for subsequent quantization, allowing the objective to capture both direct channel-induced codeword transitions and their effects on later index assignments. These updates reshape the codebook and index assignments according to channel impairments. As a result, the QAM symbol distribution changes with SNR without a target distribution or shaping loss. We analyze why this behavior appears from an information-theoretic perspective in Appendix A.

One special case is the simplified setting with neither the hyperprior nor quadtree autoregressive context, where cross-group error propagation is removed and the receiver-side codebook update is equivalent in expectation to CAVQ (Meng et al., 2025). Another special case is the noiseless setting. When the channel noise disappears,

$$N = 0, \quad h = 1 \quad \Longrightarrow \quad r = s_J, \qquad \hat{J} = J, \qquad \hat{u} = u, \qquad \mathcal{L}_{\mathrm{cb}} = \left\| \mathrm{sg}(u) - c_J \right\|^2 .$$

The objective therefore reduces to standard VQ learning, with no cross-index transition to organize the otherwise arbitrary codeword labels. Together, these special cases connect the receiver-side codebook formulation to CAVQ and standard VQ under simplified coding and channel settings, while the full formulation handles autoregressive image coding over noisy channels.

3.3 Receiver Reconstruction Estimation

The receiver decisions described in Section 3.2 can affect the residuals and context used for subsequent quantization. Since SemQAM uses earlier reconstructed latent groups to encode later ones, the transmitter must approximate the receiver's reconstructions. However, it knows the sent index J but not the detected index $\hat{J}$, so SemQAM estimates the received codeword without feedback.

For hard-decision QAM, the transition matrix is (Shannon, 1948; Farvardin, 1990),

$$W_{kj} = \mathbb{P}(\hat{J} = k \mid J = j).$$

Given a transmitted index j, SemQAM estimates the received codeword by its conditional mean,

$$\mathbb{E}[c_{\hat{J}} \mid J = j] = \bar{c}_j = \sum_k W_{kj} c_k. \tag{3}$$

We refer to $\bar{c}_j$ as the expected received codeword (Expected-RX). As the conditional mean, $\bar{c}_j$ is the minimum-MSE deterministic estimate of the received codeword based only on the transmitted index. A derivation is provided in Appendix B.3.

In the noiseless case, the transition matrix reduces to the identity matrix,

$$W = I \quad \Longrightarrow \quad \bar{c}_j = c_j.$$

Thus, Expected-RX naturally reduces to the transmitted codeword when channel is noiseless.

This estimate is then used to continue the transmitter-side RVQ. For RVQ stage l, let $u^{(l)}$ denote the transmitter residual before quantization and J_l the selected index. The next residual is

$$u^{(l+1)} = u^{(l)} - \bar{c}_{J_l}. \tag{4}$$

The same expected received codeword is used as the commitment target,

$$\mathcal{L}_{\mathrm{com}} = \left\| u^{(l)} - \mathrm{sg}(\bar{c}_{J_l}) \right\|^2, \tag{5}$$

where sg denotes stop-gradient. Using $\bar{c}_{J_l}$ in both operations keeps the transmitter residual consistent with the codeword expected to be reconstructed at the receiver. Because later residuals and context pass through nonlinear transforms and quantizers, this remains a local estimate rather than the exact expectation of subsequent receiver reconstructions.

We optimize the reconstruction with the EF-LIC image objective (Cao et al., 2026),

$$\mathcal{L}_{\mathrm{img}} = \lambda_{\mathrm{rec}} \|x - \hat{x}\|_2^2 + \lambda_{\mathrm{per}} \mathcal{L}_{\mathrm{LPIPS}}(x, \hat{x}) + \lambda_{\mathrm{adv}} \mathcal{L}_{\mathrm{GAN}}(x, \hat{x}),$$

where $\mathcal{L}_{\mathrm{LPIPS}}$ uses pretrained VGG features (Simonyan & Zisserman, 2015; Zhang et al., 2018), and $\mathcal{L}_{\mathrm{GAN}}$ is the generator-side PatchGAN loss following VQGAN (Esser et al., 2021). Together with the receiver-side codebook loss in Equation (1), the complete training objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{img}} + \lambda_{\mathrm{cb}} \mathcal{L}_{\mathrm{cb}} + \lambda_{\mathrm{com}} \mathcal{L}_{\mathrm{com}}, \tag{6}$$

where λ_{rec}, λ_{per}, λ_{adv}, λ_{cb}, and λ_{com} are the corresponding loss weights.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Training Details. SemQAM uses four CBR settings for each of 10 dB, 16 dB, 24 dB, and noiseless transmission, giving sixteen codebook groups. The noiseless variant is denoted as SemQAM-C for readability. It remains part of the same SemQAM model and shares the same weights. We train SemQAM on ImageNet (Deng et al., 2009) using a three-stage multi-operating-point training strategy. Detailed training settings are provided in Appendix C.1.

Evaluation Datasets and Metrics. We evaluate on Kodak, Tecnick, DIV2K validation, and CLIC 2020 Professional (Eastman Kodak Company, 1993; Asuni & Giachetti, 2015; Agustsson & Timofte, 2017; Toderici et al., 2020) using LPIPS, DISTS, and FID (Zhang et al., 2018; Ding et al., 2022; Heusel et al., 2017). We report CBR-based BD-rate (Bjøntegaard, 2001) to measure the average transmission-resource difference at matched perceptual quality. All methods follow the CBR definition in Section 2, with required side information and padding overhead included. Latency is measured on one NVIDIA A100, and SemQAM reports the mean CUDA latency across the evaluated CBR settings. Metric, rate, and runtime protocols are provided in Appendices C.3 to C.5.

Compared Methods. We consider two groups of baselines for complementary comparisons. (i) NTSCC, NTSCC+, MDJCM-A, and Quad-JSCC (Dai et al., 2022; Wang et al., 2023; Zhang et al., 2025; Huang & Qin, 2026) are learned JSCC methods used to evaluate SemQAM against existing joint transmission systems. (ii) MS-ILLM, OSCAR, OneDC, DiT-IC, EF-LIC, and CoD-Lite (Muckley et al., 2023; Guo et al., 2025; Xue et al., 2025b; Shi et al., 2026; Cao et al., 2026; Jia et al., 2026) are generative image codecs paired with LDPC and QAM with MCS table (3GPP, 2020) in Appendix C.4, forming separated systems for comparison of perceptual quality and transmission efficiency. Following the common evaluation setting in learned image JSCC (Dai et al., 2022; Wang et al., 2023; Huang & Qin, 2026), our main comparisons use the AWGN channel at 10 dB to enable broad comparison with existing methods. Additional comparisons at 16 dB and 24 dB and baseline-specific configurations are provided in Appendices C.3 and D.1.

4.2 COMPARISON WITH PREVIOUS METHODS

Comparison with JSCC Methods. Perceptual and generative JSCC methods remain relatively limited, and most evaluated JSCC baselines are optimized primarily for distortion rather than perceptual quality. As shown in Figure 2 and Table 1, SemQAM substantially outperforms NTSCC, NTSCC+, and MDJCM-A in LPIPS, DISTS, and FID, and also achieves clear gains over the recent perceptual Quad-JSCC while requiring much lower inference latency. Notably, conventional learned JSCC commonly allows unconstrained continuous-valued channel inputs, whose performance can be viewed as the limiting case approached by increasingly high modulation orders (Tung et al., 2022; Bo et al., 2024; Ying et al., 2026). Prior studies report that finite-constellation performance improves toward this continuous-input reference as the modulation order increases (Tung et al., 2022; Bo et al., 2024). SemQAM nevertheless achieves stronger perceptual performance while operating directly with finite 1024-QAM, demonstrating strong joint transmission capability under a practical discrete modulation constraint.

Comparison with Separated Systems. The source-coding variant SemQAM-C already performs strongly among the compared generative codecs, showing that the unified model retains strong conventional source-coding capability. At 10 dB, SemQAM further improves over the separated SemQAM-C baseline. This matched comparison demonstrates the benefit of jointly optimizing semantic coding, channel coding, and modulation under channel impairments.

4.3 CONSTELLATION USAGE ANALYSIS

Figure 3 shows that SemQAM learns strongly SNR-dependent symbol distributions without explicitly constraining either the VQ index probabilities or the QAM symbol usage. At 10 dB, the distribution concentrates on the corner symbols, which have fewer directions in which noise can cause confusion than interior symbols. Because the noise power is scaled with the average transmit-

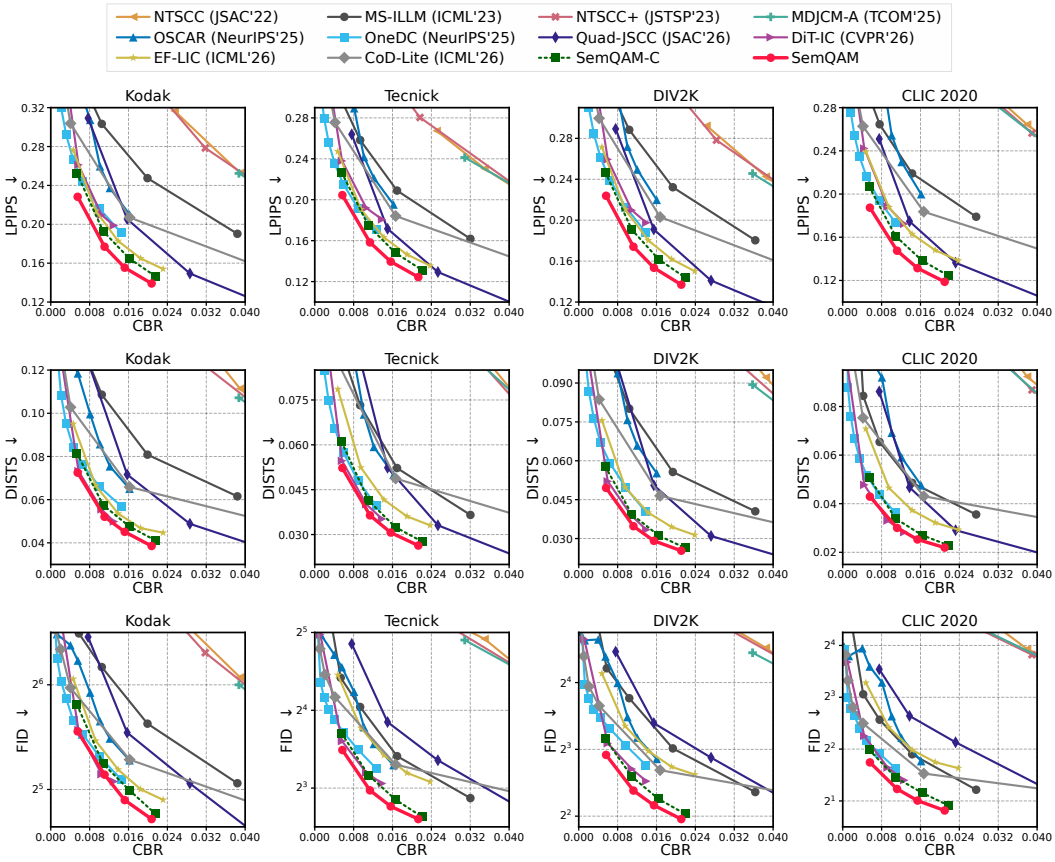


Figure 2: Rate and perceptual quality at 10 dB across four datasets. Lower LPIPS, DISTS, and FID indicate better quality. FID uses base-2 logarithmic vertical axes.

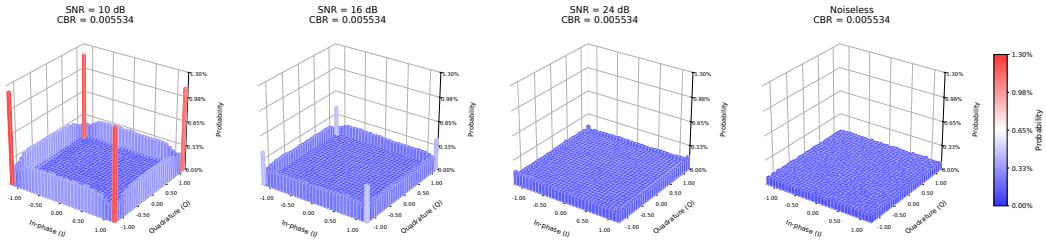


Figure 3: QAM symbol usage under different channel conditions. Lower SNR produces stronger concentration on the corner symbols, while the distribution is more balanced at higher SNR.

ted symbol power, using higher-energy corner symbols does not provide a higher operating SNR. As the SNR increases, this preference weakens and the symbol usage becomes increasingly balanced, approaching a nearly uniform distribution in the noiseless setting. This behavior appears without an explicit shaping objective, showing that symbol usage changes with SNR as a result of joint optimization. We explain this behavior with information theory in Appendix A.

4.4 Effect of SNR at a Fixed CBR

As shown in Figure 4, SemQAM consistently improves as the SNR increases from 10 dB to 16 dB and 24 dB across all four datasets. Moreover, SemQAM maintains lower LPIPS than the compared generative image codecs paired with LDPC and QAM at every evaluated SNR, showing that its advantage persists across channel conditions under the same transmission budget.

Table 1: Results on Kodak at 10 dB, with latency measured on one NVIDIA A100. Latency is reported in ms, FLOPs in G, parameters in M, and BD-rate in percent. JSCC denotes joint semantic-channel coding, Percep. denotes generative optimization, and Mod. denotes joint optimization with modulation. SemQAM-C denotes the separated system combining the noiseless source codec with LDPC and QAM, while SemQAM denotes joint QAM transmission. SemQAM-C is the BD-rate anchor, and lower values are better. Bold and underline mark the best and second-best BD-rates.

	Computation				BD-rate ↓			Design		
Method	Enc.	Dec.	FLOPs	Params	LPIPS	DISTS	FID	JSCC	Percep.	Mod.
MS-ILLM	165.38	147.79	599.52	181.40	252.63	286.55	198.99	—	✓	—
OSCAR	53.04	167.56	6485.61	1009.30	91.81	96.05	60.47	—	✓	—
OneDC	76.98	204.47	8557.18	1406.42	20.96	17.01	-12.00	—	✓	—
DiT-IC	97.62	151.89	5743.15	1069.41	18.39	<u>-11.98</u>	**-18.28**	—	✓	—
EF-LIC	17.62	13.72	279.61	35.74	12.38	14.06	16.51	—	✓	—
CoD-Lite	9.33	20.86	771.69	80.15	78.39	108.03	47.56	—	✓	—
NTSCC	36.07	30.12	533.55	31.36	473.89	791.14	548.62	✓	—	—
NTSCC+	49.77	49.23	357.06	58.59	438.79	678.05	507.44	✓	—	—
MDJCM-A	33.48	59.47	534.96	32.93	443.14	716.75	501.08	✓	—	—
Quad-JSCC	53.59	182.40	531.81	46.65	67.60	109.17	109.56	✓	✓	—
SemQAM-C	4.30	5.29	381.24	60.34	<u>0.00</u>	0.00	0.00	—	✓	—
SemQAM (10dB)	4.79	5.44	381.34	60.34	**-18.57**	**-16.91**	<u>-15.37</u>	✓	✓	✓

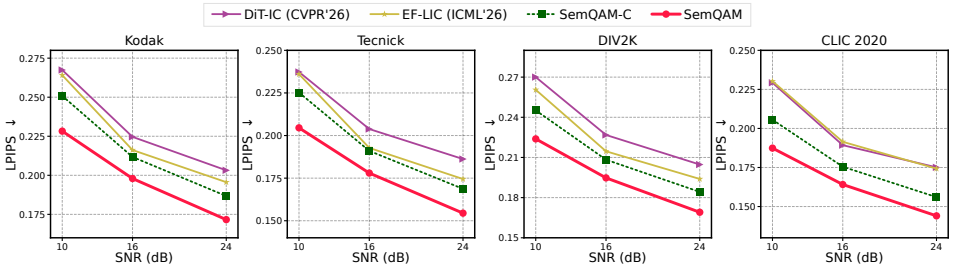


Figure 4: LPIPS at CBR 0.005534 under different SNRs across four datasets. Lower is better.

4.5 Ablation Studies

We evaluate six configurations on Kodak at 10 dB under the same training protocol for a fair comparison, as detailed in Appendix C.7. M_0 removes both the hyperprior (HP) and quadtree autoregressive context (AR), while M_1 increases the backbone capacity and M_2 restores the hyperprior. With neither the hyperprior nor quadtree autoregressive context, the receiver-side codebook loss in M_0 and M_1 is equivalent in expectation to CAVQ (Meng et al., 2025). Starting from the complete model M_5, M_3 replaces the expectation of received codewords with the transmitted codewords, while M_4 replaces the receiver-side codebook loss with a transmitter-side one.

As shown in Table 2, both additional backbone capacity and the hyperprior improve performance, while autoregressive provides a further gain, indicating that the improvement is not explained by model capacity. Compared with M_5, M_3 degrades because the transmitted codewords cannot accurately approximate the receiver reconstruction, increasing error propagation during autoregressive coding. Using a transmitter-side codebook loss in M_4 degrades performance because the transitions between transmitted and detected indices are no longer learned from receiver decisions.

5 Related Work

Learned Image JSCC. Learned image JSCC do not use digital modulation, but directly transmit continuous symbols over the wireless channel (Bourtsoulatze et al., 2019; Dai et al., 2022;

Table 2: Ablation configurations on Kodak at 10 dB. HP, AR, Deeper Backb., E-RX, and RX-CB denote the hyperprior, quadtree autoregression, deeper backbone, expected received codewords, and receiver-side codebook loss, respectively. M_0 is the LPIPS BD-rate anchor, and lower is better.

Model	Ablation method					Computation			BD-rate (%)
	HP	AR	Deeper Backb.	E-RX	RX-CB	Enc.	Dec.	FLOPs	LPIPS
M_0	—	—	—	✓	✓	2.31	3.77	310.29	0.00
M_1	—	—	✓	✓	✓	2.79	4.68	397.52	-4.84
M_2	✓	—	—	✓	✓	3.84	4.86	368.66	-5.15
M_3	✓	✓	—	—	✓	4.95	5.93	381.44	-7.99
M_4	✓	✓	—	✓	—	5.46	5.90	381.44	> 100
M_5	✓	✓	—	✓	✓	5.31	5.60	381.44	**-12.29**

Wang et al., 2023). Quad-JSCC further provides generative reconstruction (Huang & Qin, 2026). SemQAM jointly optimizes generative image transmission under a fixed finite QAM constellation.

Learned Image JSCM. DeepJSCC-Q constrains channel symbols to finite constellations, VQ-DeepSC transmits quantized features over digital channels, while JCM and MDJCM optimize symbol probabilities or multiple modulation orders (Tung et al., 2022; Fu et al., 2023; Bo et al., 2024; Zhang et al., 2025). SemQAM instead maps VQ indices directly to standard QAM symbols and trains the representation using detected indices, adapting latent representation to channel impairments.

Noisy Channel Vector Quantization. Classical channel-optimized quantizers incorporate channel errors into index assignment and codebook optimization (Farvardin & Vaishampayan, 1987; Farvardin, 1990), while CAVQ incorporates direct index transitions caused by channel errors into codebook learning (Meng et al., 2025). However, CAVQ does not model error propagation through autoregressive coding, while removing these dependencies leads to lower reconstruction performance in our experiments. EF-LIC provides the autoregressive RVQ representation (Cao et al., 2026), but assumes consistent transmitter and receiver reconstructions. SemQAM uses Expected-RX to approximate receiver reconstructions at the transmitter, improving reconstruction performance.

6 Limitations

Given finite experimental resources, we prioritize 16 operating points that allow direct comparison with representative JSCC and JSCM evaluations (Tung et al., 2022; Dai et al., 2022; Zhang et al., 2025; Huang & Qin, 2026). These operating points cover multiple CBRs and SNRs under AWGN, with the 10 dB AWGN condition used as the primary setting to align with them. However, the two core designs of SemQAM are not specific to AWGN, since codebook learning uses detected indices and Expected-RX uses the corresponding transition probabilities. Broader fading conditions and practical physical-layer settings such as OFDM may require more general transition modeling and are left for future evaluation.

7 Conclusion

We presented SemQAM, a joint semantic-channel coding and modulation system for generative image transmission with standard QAM. SemQAM trains VQ codewords using detected indices at the receiver, so transitions between transmitted and detected indices affect codebook learning, while symbol usage changes with SNR as a result of joint optimization. It further uses the expectation of received codewords to approximate receiver reconstructions and reduce error propagation during autoregressive coding. Experiments demonstrate the advantage of joint coding over separate optimization while retaining the performance and efficiency of conventional source coding, with SemQAM also outperforming the compared JSCC methods under perceptual metrics.

AI USE STATEMENT

AI tools were used to assist with manuscript organization, language refinement, and content verification. The authors reviewed all text, citations, code, calculations, claims, and artifacts produced with AI assistance against the cited sources, implementation, and experimental records. Generative AI was not used to create datasets or experimental measurements. The authors take responsibility for the final content of this work, including material produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Section 3 specifies the model architecture, channel interface, receiver-side codebook learning, Expected-RX, and training objective. Section 4 describes the experimental settings, baselines, metrics, and main evaluation results. The appendices provide the corresponding theoretical analysis, implementation details, training and evaluation protocols, and additional experimental results. The source code will be publicly released upon acceptance.

REFERENCES

3GPP. NR: Physical Layer Procedures for Data. Technical Report TS 38.214, version 16.2.0, Release 16, 3rd Generation Partnership Project, 2020. Table 5.1.3.1-1.

Eirikur Agustsson and Radu Timofte. NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1122–1131, 2017.

Nicola Asuni and Andrea Giachetti. TESTIMAGES: A Large Data Archive for Display and Algorithm Testing. *Journal of Graphics Tools*, 17(4):113–125, 2015.

Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational Image Compression with a Scale Hyperprior. In *International Conference on Learning Representations*, 2018.

Gisle Bjøntegaard. Calculation of Average PSNR Differences between RD-Curves. Technical Report VCEG-M33, ITU-T VCEG, 2001.

Yochai Blau and Tomer Michaeli. The Perception-Distortion Tradeoff. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6228–6237, 2018.

Yochai Blau and Tomer Michaeli. Rethinking Lossy Compression: The Rate-Distortion-Perception Tradeoff. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 675–685. PMLR, 2019.

Yufei Bo, Yiheng Duan, Shuo Shao, and Meixia Tao. Joint Coding-Modulation for Digital Semantic Communications via Variational Autoencoder. *IEEE Transactions on Communications*, 72(9): 5626–5640, 2024.

Eirina Bourtsoulatze, David Burth Kurka, and Deniz Gündüz. Deep Joint Source-Channel Coding for Wireless Image Transmission. *IEEE Transactions on Cognitive Communications and Networking*, 5(3):567–579, 2019.

Hao Cao, Wenqi Guo, Zhijin Qin, and Jungong Han. Efficient Learned Image Compression without Entropy Coding. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*. PMLR, 2026.

Centre Tecnològic de Telecomunicacions de Catalunya. 5G-LENA: ns3::NrEesmT1 Struct Reference. Official simulator documentation and reference implementation, 2026. URL `https://cttc-lena.gitlab.io/nr/html/structns3_1_1_nr_eesm_t1.html`. Reference curves accessed September 11, 2026.

Kyongkuk Cho and Dongweon Yoon. On the General BER Expression of One- and Two-Dimensional Amplitude Modulations. *IEEE Transactions on Communications*, 50(7):1074–1080, 2002.

Jincheng Dai, Sixian Wang, Kailin Tan, Zhongwei Si, Xiaoqi Qin, Kai Niu, and Ping Zhang. Nonlinear Transform Source-Channel Coding for Semantic Communications. *IEEE Journal on Selected Areas in Communications*, 40(8):2300–2316, 2022.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

Keyan Ding, Kede Ma, Shiqi Wang, and Eero P. Simoncelli. Image Quality Assessment: Unifying Structure and Texture Similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5):2567–2581, 2022.

Eastman Kodak Company. Kodak Lossless True Color Image Suite (PhotoCD PCD0992), 1993. URL `https://r0k.us/graphics/kodak/`.

Patrick Esser, Robin Rombach, and Björn Ommer. Taming Transformers for High-Resolution Image Synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12873–12883, 2021.

Nariman Farvardin. A Study of Vector Quantization for Noisy Channels. *IEEE Transactions on Information Theory*, 36(4):799–809, 1990.

Nariman Farvardin and Vinay A. Vaishampayan. Optimal Quantizer Design for Noisy Channels: An Approach to Combined Source-Channel Coding. *IEEE Transactions on Information Theory*, 33(6):827–838, 1987.

Qifan Fu, Huiqiang Xie, Zhijin Qin, Gregory Slabaugh, and Xiaoming Tao. Vector Quantized Semantic Communication System. *IEEE Wireless Communications Letters*, 12(6):982–986, 2023.

Robert G. Gallager. Low-Density Parity-Check Codes. *IRE Transactions on Information Theory*, 8 (1):21–28, 1962.

Deniz Gündüz, Michèle A. Wigger, Tze-Yang Tung, Ping Zhang, and Yong Xiao. Joint Source–Channel Coding: Fundamentals and Recent Progress in Practical Designs. *Proceedings of the IEEE*, 113(9):888–919, 2025.

Jinpei Guo, Yifei Ji, Zheng Chen, Kai Liu, Min Liu, Wang Rao, Wenbo Li, Yong Guo, and Yulun Zhang. OSCAR: One-Step Diffusion Codec Across Multiple Bit-rates. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

J. C. Hancock and R. W. Lucky. Performance of Combined Amplitude and Phase-Modulated Communication Systems. *IRE Transactions on Communications Systems*, 8(4):232–237, 1960.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Yinhuan Huang and Zhijin Qin. Image Semantic Communication With Quadtree Partition-Based Coding. *IEEE Journal on Selected Areas in Communications*, 44:2765–2778, 2026.

Dennis Hui, Sara Sandberg, Yufei Blankenship, Mattias Andersson, and Leefke Grosjean. Channel Coding in 5G New Radio: A Tutorial Overview and Performance Comparison with 4G LTE. *IEEE Vehicular Technology Magazine*, 13(4):60–69, 2018.

Zhaoyang Jia, Jiahao Li, Bin Li, Houqiang Li, and Yan Lu. Generative Latent Coding for Ultra-Low Bitrate Image Compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26088–26098, 2024.

Zhaoyang Jia, Bin Li, Jiahao Li, Wenxuan Xie, Linfeng Qi, Houqiang Li, and Yan Lu. Towards Practical Real-Time Neural Video Compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12543–12552, 2025.

Zhaoyang Jia, Naifu Xue, Zihan Zheng, Jiahao Li, Bin Li, Xiaoyi Zhang, Zongyu Guo, Yuan Zhang, Houqiang Li, and Yan Lu. CoD-Lite: Real-Time Diffusion-Based Generative Image Compression. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*. PMLR, 2026.

Shiyin Jiang, Wei Long, Minghao Han, Zhenghao Chen, Ce Zhu, and Shuhang Gu. Differentiable Vector Quantization for Rate-Distortion Optimization of Generative Image Compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14440–14450, 2026.

Victoria Kostina and Sergio Verdú. Lossy Joint Source-Channel Coding in the Finite Blocklength Regime. *IEEE Transactions on Information Theory*, 59(5):2545–2575, 2013.

Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. High-Fidelity Audio Compression with Improved RVQGAN. In *Advances in Neural Information Processing Systems*, volume 36, pp. 27980–27993, 2023.

David Burth Kurka and Deniz Gündüz. DeepJSCC-f: Deep Joint Source-Channel Coding of Images With Feedback. *IEEE Journal on Selected Areas in Information Theory*, 1(1):178–193, 2020.

Anqi Li, Feng Li, Yuxi Liu, Runmin Cong, Yao Zhao, and Huihui Bai. Once-for-All: Controllable Generative Image Compression with Dynamic Granularity Adaptation. In *International Conference on Learning Representations*, 2025.

Zian Meng, Qiang Li, Wenqian Tang, Mingdie Yan, and Xiaohu Ge. Channel-Aware Vector Quantization for Robust Semantic Communication on Discrete Channels. arXiv preprint arXiv:2510.18604, 2025. URL https://arxiv.org/abs/2510.18604.

David Minnen, Johannes Ballé, and George D. Toderici. Joint Autoregressive and Hierarchical Priors for Learned Image Compression. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

Matthew J. Muckley, Alaaeldin El-Nouby, Karen Ullrich, Hervé Jégou, and Jakob Verbeek. Improving Statistical Fidelity for Neural Image Compression with Implicit Local Likelihood Models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 25426–25443. PMLR, 2023.

Thomas J. Richardson, M. Amin Shokrollahi, and Rüdiger L. Urbanke. Design of Capacity-Approaching Irregular Low-Density Parity-Check Codes. *IEEE Transactions on Information Theory*, 47(2):619–637, 2001.

Claude E. Shannon. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27: 379–423, 623–656, 1948.

Claude E. Shannon. Coding Theorems for a Discrete Source with a Fidelity Criterion. In *IRE International Convention Record*, volume 7, pp. 142–163, 1959. Part 4.

Junqi Shi, Ming Lu, Xingchen Li, Anle Ke, Ruiqi Zhang, and Zhan Ma. DiT-IC: Aligned Diffusion Transformer for Efficient Image Compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 25719–25729, 2026.

Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations*, 2015.

George Toderici, Wenzhe Shi, Radu Timofte, Lucas Theis, Johannes Ballé, Eirikur Agustsson, Nick Johnston, and Fabian Mentzer. Workshop and Challenge on Learned Image Compression (CLIC2020). CVPR 2020 Workshop, 2020. URL https://www.tensorflow.org/datasets/catalog/clic.

Tze-Yang Tung, David Burth Kurka, Mikolaj Jankowski, and Deniz Gündüz. DeepJSCC-Q: Constellation Constrained Deep Joint Source-Channel Coding. *IEEE Journal on Selected Areas in Information Theory*, 3(4):720–731, 2022.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural Discrete Representation Learning. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Sixian Wang, Jincheng Dai, Xiaoqi Qin, Zhongwei Si, Kai Niu, and Ping Zhang. Improved Nonlinear Transform Source-Channel Coding to Catalyze Semantic Communications. *IEEE Journal of Selected Topics in Signal Processing*, 17(5):1022–1037, 2023.

Huiqiang Xie, Zhijin Qin, Geoffrey Ye Li, and Biing-Hwang Juang. Deep Learning Enabled Semantic Communication Systems. *IEEE Transactions on Signal Processing*, 69:2663–2675, 2021.

Naifu Xue, Zhaoyang Jia, Jiahao Li, Bin Li, Yuan Zhang, and Yan Lu. DLF: Extreme Image Compression with Dual-generative Latent Fusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19227–19236, 2025a.

Naifu Xue, Zhaoyang Jia, Jiahao Li, Bin Li, Yuan Zhang, and Yan Lu. One-Step Diffusion-Based Image Compression with Semantic Distillation. In *Advances in Neural Information Processing Systems*, volume 38, 2025b.

Ke Yang, Sixian Wang, Jincheng Dai, Xiaoqi Qin, Kai Niu, and Ping Zhang. SwinJSCC: Taming Swin Transformer for Deep Joint Source-Channel Coding. *IEEE Transactions on Cognitive Communications and Networking*, 11(1):90–104, 2025.

Jingkai Ying, Zhijin Qin, Yulong Feng, Liejun Wang, and Xiaoming Tao. Joint Semantic-Channel Coding and Modulation for Token Communications. *IEEE Transactions on Wireless Communications*, 25:8179–8193, 2026.

Guangyi Zhang, Pujing Yang, Yunlong Cai, Qiyu Hu, and Guanding Yu. From Analog to Digital: Multi-Order Digital Joint Coding-Modulation for Semantic Communication. *IEEE Transactions on Communications*, 73(6):4257–4271, 2025.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018.

APPENDIX

In the appendix, we provide the following:

- Appendix A analyzes the learned QAM symbol probabilities from an information-theoretic perspective.
- Appendix B provides additional details on the model architecture, receiver-side codebook learning, and transmitter state estimation.
- Appendix C specifies the training, datasets, baseline inference, separated transmission, metric computation, evaluation stability, and ablation protocols.
- Appendix D presents additional results across datasets and channel conditions, together with higher-resolution latency and memory measurements.
- Appendix E provides visual comparisons with previous methods.

A QAM SYMBOL PROBABILITIES FROM INFORMATION THEORY

In this section, we analyze why the symbol probabilities learned by SemQAM exhibit the constellation patterns observed in Figure 3. SemQAM neither maximizes mutual information nor explicitly constrains the QAM symbol probabilities. We therefore introduce an independent channel-side reference under the same SNR normalization used in training and testing. Let $p_j = \Pr(J = j)$ denote the probability of transmitting the j-th QAM symbol. For an input distribution p, the average transmitted symbol power is

$$P_s(p) = \sum_j p_j |s_j|^2.$$

For a target linear SNR γ, the corresponding noise power is

$$\sigma_N^2(p) = \frac{P_s(p)}{\gamma}. \tag{7}$$

The QAM constellation remains fixed, so assigning more probability to higher-energy points increases both the average transmitted power and the noise power rather than providing a higher operating SNR. Because the noise power depends on p, the hard-decision transition probabilities also depend on the candidate distribution,

$$W_{kj}(p) = \Pr\left(\hat{J} = k \mid J = j; \sigma_N^2(p)\right). \tag{8}$$

For each target SNR, we consider the input distribution that maximizes the information preserved after hard-decision reception,

$$\begin{aligned} p_\gamma^* &= \arg\max_{\{p_j\}} I_\gamma(J; \hat{J}), \\ p_j &\geq 0, \quad \sum_j p_j = 1, \end{aligned} \tag{9}$$

where

$$I_\gamma(J; \hat{J}) = \sum_j p_j \sum_k W_{kj}(p) \log_2 \frac{W_{kj}(p)}{\sum_m p_m W_{km}(p)}. \tag{10}$$

Unlike the capacity problem of a fixed discrete channel, the transition matrix changes with p through Equation (7). We therefore optimize this objective numerically over the probability simplex and use the resulting distribution only as a channel-side reference for interpreting the learned QAM symbol usage, rather than as an optimization objective or achievable-rate claim for SemQAM.

The SNR-dependent pattern in Figure 5 follows from the hard-decision geometry of finite QAM. Interior symbols have neighboring decision regions in all directions, whereas boundary and corner symbols have one or two outward directions without adjacent constellation points. This gives boundary and corner symbols a geometric advantage under strong noise. At the same time, concentrating probability on these higher-energy symbols also increases the noise power through Equation (7). The optimized distribution therefore balances this geometric robustness against the corresponding

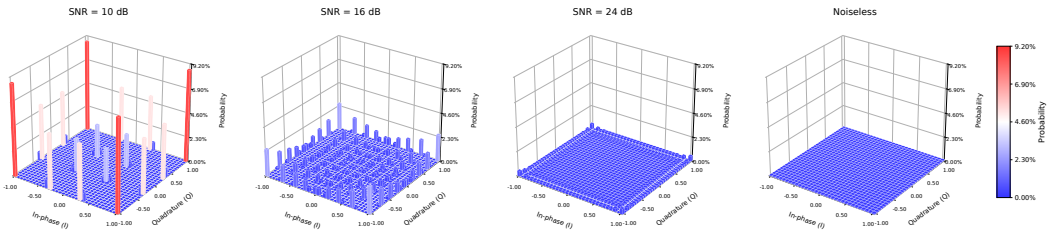


Figure 5: Numerically optimized 1024-QAM symbol usage under the same average-power SNR normalization used by SemQAM. For each candidate distribution, the noise power is determined by its average transmitted symbol power, so concentrating probability on higher-energy symbols does not increase the operating SNR. Lower SNR produces stronger concentration on corner and boundary regions, while symbol usage becomes progressively broader as SNR increases.

increase in noise and the need to use sufficiently many distinct symbols. As the channel becomes cleaner, this robustness advantage becomes less important and the distribution becomes broader. In the noiseless limit, $\hat{J} = J$ and $I(J; \hat{J}) = H(J)$, so the uniform distribution maximizes the preserved information.

The learned distributions in Figure 3 exhibit a similar large-scale SNR-dependent trend, but are considerably denser than the channel-side reference. This difference follows from the image representation requirement of SemQAM. Its symbol probabilities are not independently optimized variables, but are induced by the learned representation and nearest-neighbor VQ,

$$p_j = \Pr\left(\|u - c_j\|^2 \le \|u - c_m\|^2,\ \forall m\right). \tag{11}$$

Concentrating the representation on too few indices would reduce the number of codewords available for distinguishing image features, so the practical system must preserve sufficient codeword diversity while adapting to the QAM channel. This channel-dependent usage is induced by the receiver-side codebook learning analyzed in Appendix B.2, which changes both codeword geometry and index usage. Overall, the qualitative agreement provides an information-theoretic explanation for the SNR-dependent symbol usage observed after joint optimization.

B Additional Method Details

This section provides additional details for the model architecture, receiver-side codebook learning, and transmitter state estimation described in Section 3. We first specify the implementation inherited from EF-LIC, then analyze how receiver decisions affect codebook learning under channel errors, and finally detail how the transmitter approximates the receiver reconstructions required for subsequent quantization without feedback.

B.1 Model Architecture

SemQAM builds on EF-LIC (Cao et al., 2026), retaining its image transforms, hyperprior, four-group quadtree context model, and projected RVQ. The architecture is shown in Figure 6. The main latent and hyperlatent contain 256 and 128 channels, respectively, while the transform and context modules use a hidden width of 384. Each main-latent RVQ contains 1024 codewords in a 16-dimensional search space, and each hyperlatent RVQ contains 1024 codewords in an 8-dimensional search space.

Following the latent scaling strategy of DCVC-RT (Jia et al., 2025), we use a multiplicative normalization field q to adjust the latent representation. The hyper synthesis transform produces both q and the context features, with q constrained to be positive and clamped below at 0.5. The transmitter obtains $\tilde{q}$ from its locally reconstructed hyperlatent $\tilde{z}$, while the receiver obtains $\hat{q}$ from $\hat{z}$.

For group i, let m_i and s_i denote the context mean and scale in the normalized latent domain. The transmitter quantizes

$$\frac{y_i \oslash \tilde{q}_i - \tilde{m}_i}{\tilde{s}_i} = \frac{y_i - \tilde{\mu}_i}{\tilde{\sigma}_i},$$

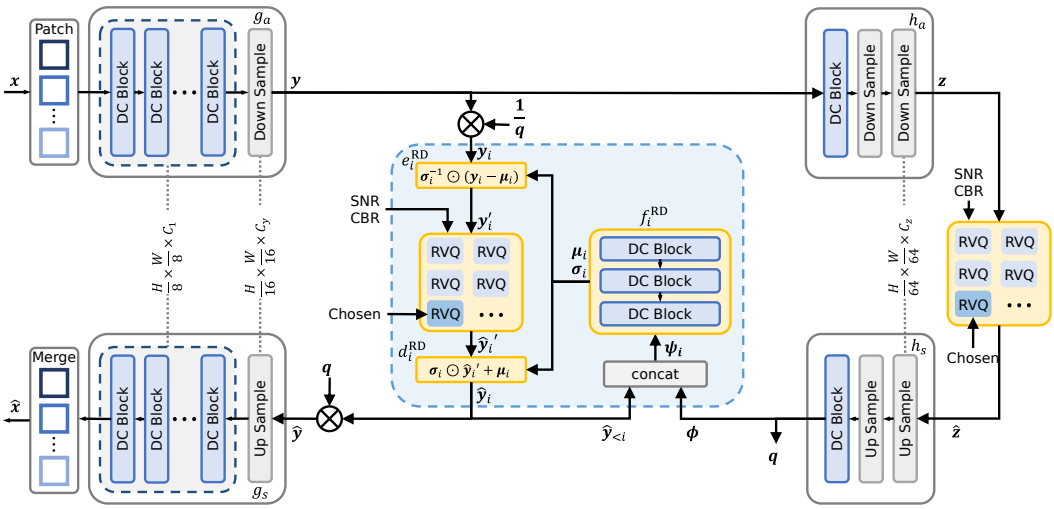


Figure 6: Framework of SemQAM. The indices produced by RVQ are mapped to standard QAM constellation points and transmitted over the wireless channel.

where the effective context parameters used in the main text are

$$\tilde{\mu}_i = \tilde{q}_i \odot \tilde{m}_i,$$
$$\tilde{\sigma}_i = \tilde{q}_i \odot \tilde{s}_i.$$

The receiver uses the corresponding locally available quantities,

$$\hat{\mu}_i = \hat{q}_i \odot \hat{m}_i,$$
$$\hat{\sigma}_i = \hat{q}_i \odot \hat{s}_i,$$

and reconstructs the group as

$$\hat{y}_i = \hat{\sigma}_i \odot \sum_l c_{\hat{j}_l} + \hat{\mu}_i.$$

Thus, the normalization field q is absorbed into the effective mean and scale in the compact formulation of Section 3.1, while the transmitter and receiver compute these parameters from their locally reconstructed hyperlatents.

Each quadtree branch combines the hyperprior support with previously reconstructed groups to predict the mean and scale of the current group. The four branches map their context features to a 384-channel hidden representation, and the shared predictor outputs a 256-channel mean and a 256-channel scale.

Role of the hyperprior and autoregressive context. The hyperprior and quadtree autoregressive context are core components of the efficient image codec considered here rather than auxiliary modules introduced for SemQAM. The hyperlatent is transmitted first and determines the normalization and context used for main-latent quantization, while each quadtree group uses previously reconstructed groups to predict the next group. These dependencies reduce coding redundancy, but they also allow an earlier detection error to change subsequent quantization.

Removing only the quadtree autoregressive context does not remove this dependency because errors in the hyperlatent can still change the quantization of the main latent. Only when both the hyperprior and quadtree autoregressive context are removed is this cross-group error propagation eliminated. In this simplified setting, the receiver-side codebook update has the same transition-weighted expectation as CAVQ (Meng et al., 2025). Removing these components therefore simplifies away the error propagation that SemQAM is designed to handle rather than providing a matched evaluation of the full image codec. For brevity, when discussing error propagation in the main text, we use *autoregressive coding* as shorthand for this context-dependent coding process involving both the hyperprior and quadtree autoregressive context.

For efficient deployment, projected RVQ is implemented using an algebraically equivalent accelerated form that reorganizes its computation without changing the selected indices or reconstructed representations.

B.2 Equivalent Form of the Receiver-Side Codebook Update Loss

For one quantization position, the receiver-side codebook update loss in Section 3.2 is

$$\mathcal{L}_{\mathrm{cb}} = \left\|\mathrm{sg}(\hat{u}) - c_j\right\|^2,$$

where $\hat{u}$ denotes the current receiver-side codebook target and $\hat{J}$ is the detected index.

For a fixed channel condition, the transition probability from transmitted index j to detected index k is

$$W_{kj} = \Pr(\hat{J} = k \mid J = j).$$

Conditioned on the current target and transmitted index $J = j$, taking the expectation over the channel decision gives

$$\begin{aligned}\mathbb{E}\left[\mathcal{L}_{\mathrm{cb}} \mid \hat{u}, J = j\right] &= \sum_k \Pr(\hat{J} = k \mid J = j) \left\|\mathrm{sg}(\hat{u}) - c_k\right\|^2 \\ &= \sum_k W_{kj} \left\|\mathrm{sg}(\hat{u}) - c_k\right\|^2. \end{aligned} \tag{12}$$

Averaging further over the training samples gives

$$\mathbb{E}\left[\mathcal{L}_{\mathrm{cb}}\right] = \mathbb{E}_{\hat{u},J}\left[\sum_k W_{kJ} \left\|\mathrm{sg}(\hat{u}) - c_k\right\|^2\right]. \tag{13}$$

Therefore, sampling the actual receiver decision and updating the corresponding codeword provides a Monte Carlo realization of the transition-weighted codebook objective in Equation (13). Under the simplified architecture described in Appendix B.1, this objective is equivalent in expectation to CAVQ (Meng et al., 2025).

Receiver-side centroid interpretation. The same objective also explains how receiver decisions shape the learned codebook. Grouping the expected loss according to the detected index gives

$$\mathbb{E}[\mathcal{L}_{\mathrm{cb}}] = \sum_k \Pr(\hat{J} = k)\, \mathbb{E}\left[\left\|\mathrm{sg}(\hat{u}) - c_k\right\|^2 \mid \hat{J} = k\right]. \tag{14}$$

For fixed assignments and detached receiver-side targets, each term in Equation (14) is a standard squared-error centroid objective. Therefore, for any detected index k with $\Pr(\hat{J} = k) > 0$, the minimizing codeword is

$$c_k^* = \mathbb{E}\left[\hat{u} \mid \hat{J} = k\right]. \tag{15}$$

Expanding this conditional mean over the transmitted index gives

$$c_k^* = \frac{\sum_j \Pr(J = j) W_{kj}\, \mathbb{E}\left[\hat{u} \mid J = j, \hat{J} = k\right]}{\Pr(\hat{J} = k)}, \tag{16}$$

where

$$\Pr(\hat{J} = k) = \sum_j \Pr(J = j) W_{kj}.$$

Thus, the receiver codeword c_k is driven by all receiver-side targets that can be detected as index k, weighted by both their transmitted-index frequency and the corresponding channel transition probability. Transmitted symbols that frequently transition to the same detected symbol therefore jointly influence the same receiver codeword, encouraging frequently confused QAM symbols to represent compatible source content.

In the noiseless limit, W becomes the identity matrix and the transmitter and receiver reconstructions coincide, so Equation (15) reduces to the standard VQ centroid. The centroid result is a conditional property of the codebook update for fixed assignments and detached targets. During end-to-end training, both the assignments and receiver-side targets evolve together with the model parameters.

B.3 Transmitter State Estimation

The receiver-side codebook loss uses the actual detected indices, which can also change the residuals and context used for subsequent quantization. During transmission, however, the transmitter knows the sent indices but cannot observe the detected indices without feedback. Expected received codewords therefore approximate the receiver reconstructions needed to continue RVQ and context-based encoding at the transmitter.

Optimality of the expected received codeword. For a transmitted index $J = j$, the expected received codeword defined in Section 3.3 is

$$\bar{c}_j = \mathbb{E}[c_{\hat{J}} \mid J = j] = \sum_k W_{kj} c_k.$$

For any deterministic prediction a based on the transmitted index, its conditional squared prediction error can be expanded around $\bar{c}_J$ as

$$\mathbb{E}\left[\left\|c_{\hat{J}} - a\right\|^2 \mid J\right] = \mathbb{E}\left[\left\|c_{\hat{J}} - \bar{c}_J\right\|^2 \mid J\right] + \left\|a - \bar{c}_J\right\|^2. \tag{17}$$

The cross term vanishes because

$$\mathbb{E}\left[c_{\hat{J}} - \bar{c}_J \mid J\right] = 0.$$

Therefore, $\bar{c}_J$ minimizes the conditional mean squared prediction error among deterministic estimates based on the transmitted index.

Transmitter and receiver reconstruction. The transmitter and receiver maintain separate reconstructions during quantization. For main latent group i, the initial residuals are

$$u^{(1)} = (y_i - \tilde{\mu}_i) \oslash \tilde{\sigma}_i,$$
$$\hat{u}^{(1)} = (y_i - \hat{\mu}_i) \oslash \hat{\sigma}_i.$$

At RVQ stage l, the transmitter selects the index J_l by nearest-neighbor search and updates its residual using the expected received codeword,

$$u^{(l+1)} = u^{(l)} - \bar{c}_{J_l}. \tag{18}$$

The receiver instead updates its residual using the actually detected codeword,

$$\hat{u}^{(l+1)} = \hat{u}^{(l)} - c_{\hat{J}_l}. \tag{19}$$

The same transmitter and receiver recursion is applied to the hyperlatent, whose local reconstruction determines the hyperprior features at each endpoint. For the main latent, the transmitter uses its predicted earlier groups $\tilde{y}_{<i}$, while the receiver uses its actual reconstructions $\hat{y}_{<i}$. Expected received codewords therefore provide the transmitter-side quantities required by both dependencies without receiver feedback.

The commitment loss in Equation (5) follows the same transmitter-side prediction. Its stop gradient keeps the expected codeword fixed under this term, while codeword learning is driven by the receiver-side codebook update loss.

C Training and Evaluation Protocol

This section specifies the training and evaluation protocols used throughout the experiments. It covers multi-operating-point optimization, dataset preprocessing, baseline inference settings, separated transmission, metric computation, channel-noise stability, and the matched protocol used for the ablation study.

C.1 Training Protocol

Multi-operating-point training. SemQAM supports four CBR settings for each of 10 dB, 16 dB, 24 dB, and noiseless transmission. Each SNR–CBR pair defines one operating point with its own

VQ codebooks, while the image transforms, hyperprior, and context networks are shared across all sixteen operating points. To stably optimize these operating points, we adopt a three-stage training strategy on ImageNet using one NVIDIA RTX 4090. First, the shared model is trained for 0.3M iterations at a learning rate of 10^{-4}, randomly selecting one of the four 10 dB operating points for each batch. Second, the shared backbone is frozen and each of the remaining twelve operating points is trained for 0.06M iterations at 10^{-4}. Finally, the complete model is jointly fine-tuned for 1M iterations at 10^{-5}, randomly selecting one of all sixteen operating points for each batch. This staged schedule provides stable optimization across multiple operating points and is not required by the model architecture. The reported models use a random seed of 1926.

Initialization stability. We further train SemQAM using five random seeds that independently initialize both the network parameters and VQ codebooks. The resulting models converge to similar QAM shaping patterns and final performance. This behavior is consistent with the permutation symmetry of VQ, where the initial codeword indices are arbitrary labels. Receiver-side codebook learning subsequently organizes these labels according to the channel transition structure, making the converged shaping behavior insensitive to the initial codeword ordering.

C.2 Datasets and Preprocessing

We evaluate SemQAM on Kodak, Tecnick, DIV2K validation, and CLIC 2020 Professional. Kodak contains 24 images at 512×768 or 768×512, and Tecnick contains 100 images at 1200×1200. We use all 100 DIV2K validation images and 250 CLIC 2020 Professional images at their original resolutions. Images are processed as RGB without resizing. When padding is required for inference, the transmitted representation includes the padded positions, while the reconstruction is cropped to the original image size before quality evaluation. The original image dimensions are used in the CBR denominator.

C.3 Baseline Inference Protocol

Learned JSCC methods. For NTSCC, NTSCC+, and MDJCM-A, we use the official publicly released implementations and checkpoints provided by the original authors. Quad-JSCC does not publicly release the corresponding perceptual-quality model, so we directly contacted the authors and corresponding author and obtained the LPIPS-optimized weights used in our evaluation. Each method is evaluated through its original transmission interface, with all required transmitted side information included in CBR.

NTSCC uses four officially released distortion-optimized operating points, while NTSCC+ uses five released operating points with fixed model weights and no online model optimization. The MDJCM authors publicly release only MDJCM-A. This analog variant serves as the upper-performance reference of the MDJCM family, so using MDJCM-A provides a strong comparison rather than a favorable setting for SemQAM. Quad-JSCC is evaluated using the perceptual-quality model obtained from its authors.

For rate accounting, NTSCC includes the transmitted main symbols and the prescribed rate-index side cost. NTSCC+ additionally transmits its side representation through the official PNG coding interface with rate-$2/3$ LDPC and 16-QAM protection. For Quad-JSCC, active real-valued main features are paired into complex channel symbols, while its encoded side information uses the selected 10 dB MCS. MDJCM-A is evaluated through its official analog transmission interface.

Separated generative codecs. The separated source-codec baselines include MS-ILLM, OSCAR, OneDC, DiT-IC, EF-LIC, and CoD-Lite. For all methods, we use the officially released implementations and pretrained checkpoints and follow their provided coding and inference procedures. Their measured source rates, including the required source side information, are converted to channel uses using the common separated-transmission protocol in Appendix C.4. The noiseless SemQAM source codec is evaluated through the same conversion to provide a matched separated reference.

For runtime evaluation, each baseline retains the numerical precision, inference backend, and acceleration strategy provided by its official implementation. For example, OneDC follows its official path with BF16 autocast for the feedforward and codec networks while retaining the large VAE in FP32, and CoD-Lite follows its official BF16 inference path. Other methods likewise use their

Table 3: Modulation and coding settings used for the separated systems. The 16 and 24 dB settings satisfy a 1% BLER threshold in the reference curves.

SNR	MCS index	QAM order	Code rate	CBR from bpp b
10 dB	16	16	$658/1024$	$b/7.7109375$
16 dB	21	64	$616/1024$	$b/10.828125$
24 dB	28	64	$948/1024$	$b/16.6640625$

released implementation settings rather than being rewritten under a common backend. This protocol evaluates the realized performance of each available codec implementation while preserving its intended execution path.

For CoD-Lite, we use the official 0.0312 bpp checkpoint and configuration. We observe that measurements from the released implementation differ from the latency values reported in its paper. At the 1024×1920 reference resolution used by CoD-Lite, the official checkpoint takes 39.52 ms for encoding and 56.86 ms for decoding on our NVIDIA A100 evaluation, compared with the reported 16 ms and 24 ms. We therefore use measurements from the released code under our common hardware and timing protocol rather than mixing them with separately reported values. Under the unified 1080×1920 evaluation used in our high-resolution comparison, CoD-Lite takes 42.37 ms for encoding and 60.47 ms for decoding, whereas SemQAM takes 15.49 ms and 22.71 ms, corresponding to $2.74\times$ and $2.66\times$ speedups, respectively.

SemQAM is evaluated with its optimized CUDA implementation under the same principle. Its CUDA acceleration is part of the proposed codec implementation, while competing methods retain the optimization paths supplied by their official releases. We report parameter count and FLOPs separately as architecture-level costs, and measured latency as complete implementation-level performance.

C.4 Separated Transmission Protocol

LDPC protection and CBR conversion. The separated systems use the modulation and coding entries in 3GPP TS 38.214, version 16.2.0, Table 5.1.3.1-1 (3GPP, 2020). Each operating point specifies a modulation order M and nominal channel-code rate R. For a source rate of b bits per pixel, the corresponding channel bandwidth ratio is

$$\mathrm{CBR} = \frac{b}{3R \log_2 M}.$$

The source payload and all source-side information required for reconstruction are included in b.

MCS selection. The 10 dB evaluation uses MCS 16. For 16 and 24 dB, we use the available 5G-LENA reference BLER curves (Centre Tecnològic de Telecomunicacions de Catalunya, 2026) and select the highest nominal spectral efficiency whose BLER does not exceed 0.01 at the evaluated SNR. The requested code block size is 4096 bits. At 16 dB, MCS 21 uses the 4096-bit reference curve and has a tabulated BLER of 0.0002. At 24 dB, MCS 28 uses the available 928-bit curve under the reference lookup rule and has a tabulated BLER of 0.002. These settings are selected directly from the reference curves without SNR interpolation.

The separated evaluation uses the reconstruction obtained after successful protected-bitstream decoding. The selected LDPC rate therefore determines the required channel resources, while image quality is evaluated from the recovered source representation.

C.5 Evaluation Protocol and Metric Computation

Choice of the 10 dB operating point. We use 10 dB as the primary channel condition for comparison with learned JSCC methods. A fixed SNR of 10 dB has been widely used for rate–quality evaluation in representative image JSCC systems, including DeepJSCC-f, DeepJSCC-Q, NTSCC, NTSCC+, SwinJSCC, MDJCM, and Quad-DeepSC (Kurka & Gündüz, 2020; Tung et al., 2022; Dai

et al., 2022; Wang et al., 2023; Yang et al., 2025; Zhang et al., 2025; Huang & Qin, 2026). In particular, these works commonly evaluate reconstruction quality across different bandwidth or CBR settings under this channel condition. All learned JSCC baselines included in our main comparison also provide an available official checkpoint or author-provided model at 10 dB, whereas model availability at other SNRs is incomplete across the compared methods. We therefore use 10 dB for the main comparison to maximize baseline coverage while remaining aligned with established JSCC evaluation settings. SemQAM additionally supports 16 dB and 24 dB, for which we report separate results to evaluate its performance under other channel conditions.

Objective perceptual evaluation. We use standardized objective perceptual metrics as the primary quantitative evaluation. Although subjective user studies can provide complementary perceptual evidence, their results depend on the participant population, viewing conditions, presentation protocol, and selected test cases, and naturally exhibit inter-observer variability. Moreover, repeated evaluations may produce different scores even with the same participants and experimental conditions, making subjective results difficult to reproduce exactly. These factors make subjective scores less suitable for reproducible and fine-grained comparisons across many methods, datasets, and operating points. In contrast, standardized objective metrics provide repeatable measurements under a fixed evaluation protocol. This practice is also widely adopted in generative image compression: representative published methods including MS-ILLM, GLC, Control-GIC, DLF, OneDC, OSCAR, and RDVQ primarily evaluate perceptual reconstruction using objective metrics such as LPIPS, DISTS, and FID without reporting a subjective user study (Muckley et al., 2023; Jia et al., 2024; Li et al., 2025; Xue et al., 2025a;b; Guo et al., 2025; Jiang et al., 2026). We therefore follow this established evaluation practice and use LPIPS and DISTS to measure reference-based perceptual fidelity and FID to measure distribution-level realism, together with qualitative reconstructions for visual inspection.

Perceptual metrics. We evaluate perceptual reconstruction using LPIPS with the VGG feature network (Zhang et al., 2018; Simonyan & Zisserman, 2015) and DISTS (Ding et al., 2022). Both metrics are computed for each reconstructed image and then averaged over the complete dataset.

Perceptual versus distortion-oriented evaluation. SemQAM targets generative perceptual reconstruction rather than distortion-optimal reconstruction. The established perception–distortion tradeoff shows that improving perceptual quality can require increased pixel-level distortion, and its rate-constrained extension further establishes a rate–distortion–perception tradeoff for lossy compression (Blau & Michaeli, 2018; 2019). PSNR therefore measures a different optimization objective and is not used as the primary criterion for the perceptually optimized setting considered here.

This distinction is particularly relevant to the learned JSCC baselines. Representative methods such as NTSCC and NTSCC+ are primarily optimized and reported under distortion-oriented objectives and PSNR-based evaluation (Dai et al., 2022; Wang et al., 2023). A PSNR comparison with these models would therefore mainly reflect different operating points on the perception–distortion tradeoff rather than provide a matched evaluation of generative perceptual reconstruction. To retain a direct comparison within the JSCC family under a perceptual objective, we additionally evaluate Quad-JSCC using the LPIPS-optimized model provided by its authors (Huang & Qin, 2026). Together with the generative image compression baselines, this provides comparisons with methods optimized toward the same perceptual objective as SemQAM.

FID. Following the patch-based FID evaluation protocol of GLC (Jia et al., 2024), we compute FID over local image patches rather than directly over the full images. We use a patch size of 64×64 for Kodak and 256×256 for the higher-resolution Tecnick, DIV2K, and CLIC 2020 datasets. For each dataset and operating point, patches from all reference images and their reconstructions are accumulated to estimate the corresponding feature distributions, from which FID is computed using Inception features. This patch-based protocol increases the number of feature samples and provides a more stable distribution-level evaluation on small image test sets. The figures use a base-two logarithmic vertical axis for visualization, while all numerical comparisons and BD-rate calculations use the original FID values.

BD-rate. We use BD-rate (Bjøntegaard, 2001) to summarize the average difference in channel uses between two rate–quality curves at matched reconstruction quality. For each comparison, one method is selected as the anchor and is therefore assigned a BD-rate of 0.00%. We interpolate $\log_{10}(\mathrm{CBR})$ as a function of the original quality metric using piecewise cubic Hermite interpolation (PCHIP) over the common quality interval. Let $a(q)$ and $b(q)$ denote the interpolated log-CBR curves of the anchor and comparison method over $[q_0, q_1]$. The reported BD-rate is

$$\text{BD-rate} = 100 \left[10^{\frac{1}{q_1 - q_0} \int_{q_0}^{q_1} (b(q) - a(q))\, dq} - 1 \right].$$

A negative BD-rate means that the comparison method requires fewer channel uses than the anchor to achieve the same quality on average, while a positive value means that it requires more. For example, a BD-rate of -20% indicates approximately 20% fewer channel uses at matched quality. The main comparisons use SemQAM-C with LDPC and QAM as the anchor. Only measured points within a valid common quality interval are used for interpolation, and comparisons without a valid interval are reported as unavailable.

C.6 Evaluation Stability

We evaluate the sensitivity of the final model to AWGN realizations by fixing the model weights, inputs, operating points, and transmission protocol and varying only the channel noise. The experiment covers all 24 Kodak images, all sixteen operating points, and ten AWGN seeds, giving 3840 reconstructions. For each operating point and seed, we first average the metric over the 24 images and then compute the sample standard deviation across the ten dataset-level means. At 10 dB and CBR 0.005534, the standard deviations are 0.0219 dB for PSNR and 0.000463 for LPIPS. At CBR 0.020752, the corresponding values are 0.0087 dB and 0.000165. These small variations show that the reported dataset-average performance is stable across the evaluated channel realizations. The noiseless operating points have zero variation.

C.7 Ablation Protocol

Configurations. The ablation configurations follow Table 2 in the main paper. M_0 removes both the hyperprior and quadtree autoregression. M_1 uses the same configuration with a deeper backbone, while M_2 restores the hyperprior without quadtree autoregression. Because removing the hyperprior changes the transmitted index count, M_0 and M_1 use their own measured CBR values.

The complete configuration M_5 uses receiver-side codebook learning and Expected-RX. Relative to M_5, M_3 replaces the expected received codewords used by the transmitter with the transmitted codewords, while M_4 replaces the receiver-side codebook loss with the transmitter-side codebook loss.

Training protocol. All ablation configurations are trained only on the four 10 dB operating points. Each configuration is trained for 1M iterations using the same dataset and optimization protocol. The four CBR points used for each ablation curve are obtained from one trained checkpoint.

D Additional Experimental Results

This section extends the main experiments along two dimensions. We first examine whether the transmission gains generalize across datasets and channel conditions, and then evaluate whether the low computational cost of SemQAM is retained at higher image resolutions.

D.1 Additional Datasets and Channel Conditions

Table 4 reports 10 dB BD-rates for representative JSCC and GIC methods across all four datasets. Relative to the matched SemQAM-C baseline, SemQAM achieves negative BD-rates in LPIPS, DISTS, and FID on every dataset. The gains are particularly consistent in LPIPS, remaining close to 19% across Kodak, Tecnick, DIV2K, and CLIC 2020. This consistency shows that the benefit of native joint training is not specific to the Kodak test set, but generalizes across image datasets with different resolutions and content. Missing entries indicate comparisons without a valid common quality interval for interpolation.

Table 4: 10 dB BD-rate (%) relative to SemQAM-C. Lower is better. Bold and underline mark the two best results. A dash denotes an unavailable comparison.

Metric	Family	Method	Kodak	Tecnick	DIV2K	CLIC
LPIPS	JSCC	NTSCC	473.89	484.29	453.53	—
		NTSCC+	438.79	472.61	413.23	914.55
		MDJCM-A	443.14	446.27	396.04	845.35
		Quad-JSCC	67.60	35.57	45.29	53.43
	GIC	MS-ILLM	252.63	143.97	195.83	227.06
		OneDC	20.96	<u>-1.14</u>	15.00	15.99
		DiT-IC	18.39	20.53	27.30	32.08
		EF-LIC	12.38	8.93	14.02	35.45
		CoD-Lite	78.39	71.48	79.93	126.14
	SemQAM	SemQAM-C	<u>0.00</u>	0.00	<u>0.00</u>	<u>0.00</u>
		SemQAM (10dB)	**-18.57**	**-20.09**	**-18.95**	**-19.59**
DISTS	JSCC	NTSCC	791.14	823.65	934.29	—
		NTSCC+	678.05	763.39	774.63	—
		MDJCM-A	716.75	769.40	801.58	—
		Quad-JSCC	109.17	83.54	94.88	86.96
	GIC	MS-ILLM	286.55	132.83	238.20	158.01
		OneDC	17.01	1.16	32.53	6.70
		DiT-IC	<u>-11.98</u>	<u>-13.28</u>	<u>-6.42</u>	**-21.97**
		EF-LIC	14.06	31.98	35.28	55.13
		CoD-Lite	108.03	111.37	120.56	164.10
	SemQAM	SemQAM-C	0.00	0.00	0.00	0.00
		SemQAM (10dB)	**-16.91**	**-18.50**	**-18.38**	<u>-18.32</u>
FID	JSCC	NTSCC	548.62	—	—	—
		NTSCC+	507.44	950.43	1085.64	—
		MDJCM-A	501.08	892.16	990.67	—
		Quad-JSCC	109.56	172.71	211.53	249.27
	GIC	MS-ILLM	198.99	110.86	182.12	115.63
		OneDC	-12.00	21.60	58.11	27.83
		DiT-IC	**-18.28**	<u>-5.85</u>	8.31	10.14
		EF-LIC	16.51	81.07	111.60	161.32
		CoD-Lite	47.56	93.28	90.15	86.47
	SemQAM	SemQAM-C	0.00	0.00	<u>0.00</u>	<u>0.00</u>
		SemQAM (10dB)	<u>-15.37</u>	**-20.47**	**-20.97**	**-23.97**

Figures 7 and 8 further extend the comparison to 16 and 24 dB using the MCS settings in Table 3. Across these additional channel conditions, SemQAM continues to provide competitive perceptual reconstruction through the same standard QAM interface. More importantly, the native SemQAM curves remain consistently improved over the matched SemQAM-C baseline, showing that adapting the source representation to the channel remains beneficial beyond the 10 dB operating point. Together, these results indicate that the gains from joint coding are stable across both image datasets and channel conditions.

D.2 Higher Resolution Execution

We further evaluate codec latency and memory at 1080p, 1440×2560, and 4K resolutions in Table 5. The SemQAM entries average all sixteen operating points at each resolution, and all methods use batch size one on A100 with their respective inference implementations.

SemQAM maintains the lowest encoding and decoding latency among the compared methods at every evaluated resolution. This advantage is preserved as the image size increases rather than being

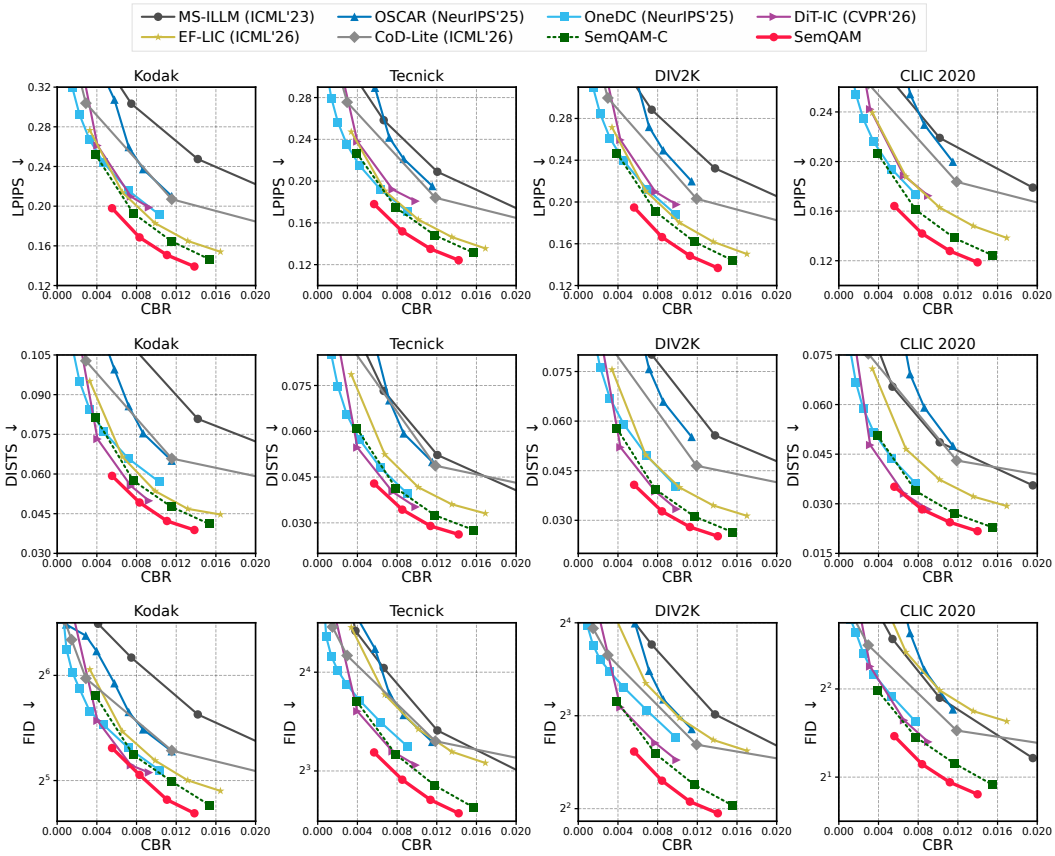


Figure 7: Additional comparison at 16 dB. Rows are LPIPS, DISTS, and FID, and columns are Kodak, Tecnick, DIV2K, and CLIC 2020 Professional. Separated codecs use MCS 21 and ideal successful decoding. FID is displayed on a base-2 logarithmic axis.

Table 5: Higher resolution latency and peak GPU memory on A100. Enc. and Dec. are in ms and peak memory is in GiB. SemQAM averages all sixteen CUDA operating points. OSCAR exceeds 80 GiB at 4K.

	1080×1920			1440×2560			2160×3840		
Method	Enc.	Dec.	Mem.	Enc.	Dec.	Mem.	Enc.	Dec.	Mem.
NTSCC	130.97	138.18	7.24	245.81	371.34	12.95	1034.13	1368.10	28.55
NTSCC+	222.02	181.14	3.96	395.41	320.84	6.85	849.50	676.75	14.28
MDJCM-A	141.46	279.99	7.69	250.12	499.22	13.57	753.93	1103.39	28.69
Quad-JSCC	158.77	267.91	4.98	273.26	586.64	13.18	597.85	1076.13	13.58
MS-ILLM	350.85	379.01	2.99	516.47	601.18	4.87	1305.93	1613.82	9.94
OSCAR	513.20	1123.38	24.75	2382.55	2719.79	79.44	—	—	> 80
OneDC	303.33	742.06	12.81	585.72	1554.82	18.86	1542.18	4846.63	35.38
DiT-IC	480.22	750.38	10.98	718.17	1111.59	14.47	1581.91	2448.49	25.41
EF-LIC	19.53	35.62	0.65	31.58	55.50	1.04	65.24	116.28	2.10
CoD-Lite	42.37	60.47	2.45	73.61	105.09	3.97	163.43	236.05	8.33
SemQAM	15.49	22.71	0.85	25.10	38.13	0.88	55.86	87.18	0.97

limited to the relatively small benchmark images used in the main experiments. At 4K, SemQAM requires 55.86 ms for encoding and 87.18 ms for decoding, while its peak GPU memory remains

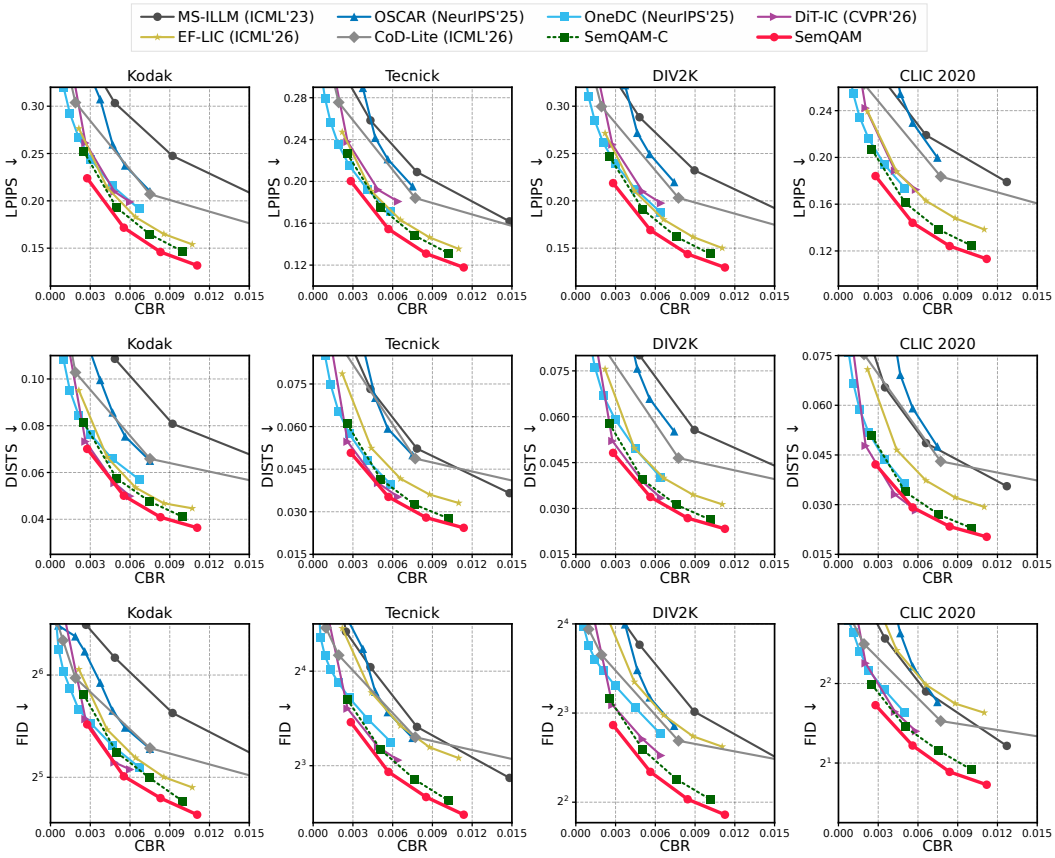


Figure 8: Additional comparison at 24 dB with the same metric and dataset arrangement as Figure 7. Separated codecs use MCS 28 and ideal successful decoding. SemQAM uses the codebooks for the corresponding 24 dB operating points.

below 1 GiB. The modest increase in memory together with the sustained latency advantage shows that the lightweight design scales efficiently to high-resolution image transmission.

E Visual Comparison

Figures 9 and 10 provide additional visual comparisons with previous methods. In all selected examples, SemQAM achieves both lower CBR and lower LPIPS than the compared methods while preserving more recognizable local structures.

In Figure 9, the enlarged regions show differences in thin lines and regular structures. For the butterfly image, SemQAM preserves the thin vein inside the wing with a continuous shape, while some compared methods blur or simplify this structure. For the road image, the faint yellow center line remains visible in the SemQAM reconstruction, whereas it becomes weaker or disappears in several compared results. For the building image, SemQAM retains the horizontal lines of the window blinds, while these lines are less distinguishable in the other reconstructions. A similar difference can be observed in the castle image, where the divisions inside the window remain clearer with SemQAM.

Figure 10 further compares SemQAM with recent generative image codecs. For the glasses, the nose pad reconstructed by SemQAM remains close to the elliptical shape and internal structure in the original image, while OneDC reconstructs it as a simpler open ring. For the railing, SemQAM preserves the repeated X patterns and their connections, whereas CoD-Lite and OSCAR show stronger distortion and merging between the bars. The enlarged sculpture region also shows that SemQAM retains the curved boundaries and separation between neighboring structures more clearly. These



Figure 9: Visual comparison with previous methods. LPIPS and CBR are reported for each reconstruction, and the marked regions are enlarged for comparison.

examples show that the lower LPIPS of SemQAM is accompanied by better preservation of object parts, thin structures, and regular geometric patterns while using fewer channel resources.



Figure 10: Additional visual comparison with previous methods. LPIPS and CBR are reported for each reconstruction, and the marked regions are enlarged for comparison.