

The Impact of Wireless Channel Errors on the Visual Quality of Learning-Based Image Coding

Ablah AlAmri ^{1,3} and Charith Abhayaratne ^{1,2}

¹School of Electrical and Electronic Engineering, The University of Sheffield, Sheffield, S1 3JD, United Kingdom.

²Centre for Machine Intelligence, The University of Sheffield, Sheffield, S10 2TN, United Kingdom.

³King Abdulaziz University, Jeddah, Saudi Arabia

Abstract. Recently, learning-based image codecs have improved compression leading to excellent performance in bitrate reduction. However, their performance when transmitted over lossy channels has not been well studied. This paper investigates how these learning-based image codecs perform under lossy channel transmission conditions. For this, we set up an experimental model that includes an encoder, a channel coding module, a channel simulation module and a decoder to evaluate the visual quality performance under various channel conditions. We compare the performance of several artificial intelligence models with the standard JPEG under various channel conditions and across various bitrates. The experimental results show that, under clean conditions, the learning-based codecs used in the experiments outperform JPEG in terms of PSNR and MS-SSIM. However, in a noisy channel, these codecs show significant degradation in PSNR and MS-SSIM under low-SNR conditions (especially below 12 dB SNR), whereas JPEG is more robust to channel errors and shows a more gradual degradation in quality as the SNR decreases.

Keywords: channel errors, error robustness, JPEG AI, visual quality, learning-based image coding, AI-based image coding.

1 Introduction

Next generation networks (6G) [1] integrate space, air, ground, and sea to enable the Internet of Everything (IoE). This environment generates massive amounts of data that can be used to train artificial intelligence (AI) models. Thus, efficient compression is crucial for reducing bandwidth usage, storage requirements, and latency. Classical image compression frameworks involve several independent stages [2], as shown in Fig. 1(a). On the transmitter side, the process starts with the input image. The source encoder first applies a transform, such as the discrete cosine transform (DCT) in JPEG [3] or the discrete wavelet transform (DWT) in JPEG2000 [4], to convert the image into a compact representation. Then, the compact representation is quantised and mapped into a compressed bitstream using Huffman coding (JPEG) or arithmetic coding (JPEG2000), which is then channel encoded and sent over a channel. On the receiver side, the channel decoder converts the received data back into the compressed bitstream. The source decoder performs entropy decoding, inverse quantisation, and inverse transform to reconstruct the image. The new learning-based image coding frameworks integrate neural networks (NNs) [5] into compression pipelines to achieve higher compression, as shown in Fig. 1 (b). Some methods integrate the NNs for a specific reason into one stage, such as using CNNs as prediction tools for missing pixels [6], or replacing one of the stages of the compression pipelines with a new NN component for source encoder and decoder. For example, [5] used NNs for downsampling and upsampling before using the JPEG codec. However, in these methods,

only the NN components (e.g. image encoder and decoder) are trainable, as the other parts of the pipelines are non-differentiable and can not be trained (e.g. quantisation and entropy coder). The use of NNs has been expanded not only to the encoder and decoder parts but also to entropy coding. Instead of relying on fixed Huffman or arithmetic coding tables, the authors in [7] introduced a learned entropy model that allows for an end-to-end (E2E) training model with a neural autoencoder. The model uses a uniform noise addition during training instead of the hard quantisation step, estimates the probability distribution of the quantised latents using NNs, and converts the compressed latents into the final bitstream using a range encoder.

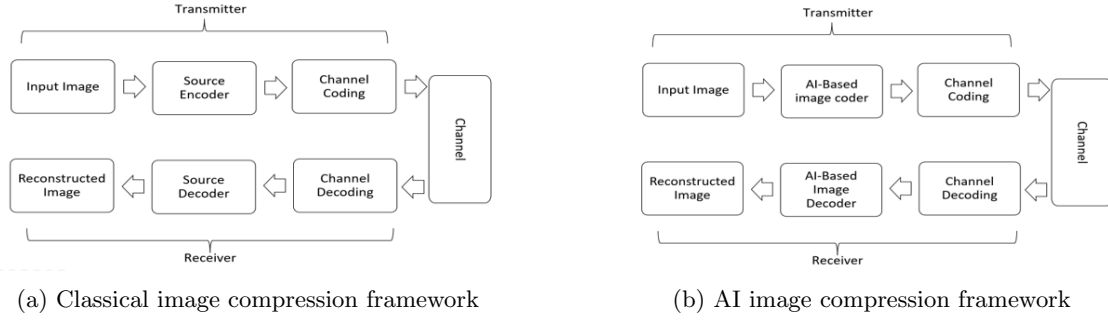


Fig. 1: Image compression pipelines: (a) classical image compression model and (b) learning-based image compression. The main difference is the second component, in which the classical source encoder/decoder is replaced with a learning-based source encoder/decoder.

Recently, the JPEG Committee introduced JPEG AI [8], a learning-based image codec framework that uses E2E deep learning models to encode images into low-bitrate bitstreams while preserving perceptual quality. The transmitter components of this framework are similar to those used in learned entropy models [7, 9, 10]. The main difference is that on the receiver side, instead of only reconstructing the input image and performing image processing tasks (e.g., super-resolution, denoising, colour correction, and inpainting), this model can perform computer vision tasks (e.g. image classification & object detection and recognition) directly on the bitstreams without reconstructing the image. As JPEG AI will be the new standard, its robustness in real-world environments is essential.

As wireless communication systems evolve, image transmission faces challenges due to channel conditions, which creates the need for efficient compression and error correction [11–15]. Traditional systems send raw or compressed data (bitstreams) over the channel, while new systems such as semantic systems and E2E joint source–channel coding (JSCC) systems, are send only features or maps that reduce the bandwidth [16–19]. However, to the best of our knowledge, most existing research on learning-based codecs (LBCs) has focused on rate-distortion (R-D) performance in optimal transmission scenarios. The effect of wireless transmission errors on AI-based image bitstreams has not been well studied. Traditional communication systems are the appropriate choice in this study because they directly transmit the compressed bitstream generated by AI-based models. Semantic and joint source–channel coding systems do not convert latents into bitstreams; thus they are not suitable for testing the robustness of image codecs based on the JPEG AI framework.

In this paper, we analyse the effect of transmission errors over wireless channels on the visual quality of images encoded with LBCs. We focus on state-of-the-art models implemented in the open-source CompressAI library [20], which provides E2E neural image compression models similar to those used in JPEG AI. The main contributions of this paper are summarised as follows:

- We analyse the effect of wireless channel errors on the visual quality of images compressed with LBCs, focusing on state-of-the-art models implemented in the CompressAI library.
- We compare the RD performance across all tested methods and illustrate the performance over different signal-to-noise ratio (SNR) ranges and bitrates using the Kodak dataset [21] to understand their robustness to transmission errors.
- We provide detailed insights into the reasons for this vulnerability.

The rest of the paper is organised as follows: Section 2 discusses various learning-based image compression methods and the wireless communication systems. Section 3 describes the methodology. Section 4 presents the simulation results of the deep learning methods under various channel conditions. These results are discussed in Section 5, followed by the conclusion and future directions in Section 6.

2 Related works

This review of the literature focuses on two main areas: learning-based image compression methods and image transmission over wireless networks. A detailed discussion of these methods and the available wireless communication systems is provided below.

Learning-Based Image Compression This method integrates NNs into specific stages of compression pipelines. For example, CNNs were used as prediction tools for missing pixels [6], while the authors in [5] used NNs for downsampling and upsampling before using the JPEG codec. A symmetric convolutional autoencoder developed by [22] trains the model with uniform noise for quantisation steps to simulate quantisation effects and uses principal component analysis and the JPEG2000 entropy coder. Other studies explored semantic compression in which CNNs identify multiple regions of interest and apply adaptive JPEG compression based on saliency levels [23]. Generative models [24] were used to develop GAN-based methods for low-bitrate compression and conditional selective generative compression (SC), which ensures that critical regions are preserved in high detail while less important areas are synthesised using a semantic label map. However, using a non-trainable standard entropy coder prevents the full E2E optimisation of the RD trade-off. In addition, generative compression approaches [24] were used to improve visual quality at low bitrates using perceptual methods, which have high computational costs and may introduce artefacts in less interesting regions. Semantic compression [23] provided adaptive region-based control but introduced complexity in saliency estimation and parameters used for different parts of the image.

JPEG AI framework The Joint Photographic Experts Group (JPEG) has been working on a new standard, JPEG AI [8]. It is completely based on AI to produce compact latent representations. It uses an entropy encoder to produce bitstreams that are sent over a channel to the receiver, which may perform entropy decoding to get back the latent and apply standard reconstruction. The earliest learned image compression that was fully

E2E optimised was Factorized Prior [7], which introduced a variational autoencoder that assumed that the elements of latent variables were independent, but this limited the compression performance. To overcome this, the Scale Hyperprior was introduced, which uses a separate learning latent space to predict the scale (variance) of the latents. This improved the rate distortion, but the model only used variance, which did not capture the full dependencies. It was expanded by incorporating a Context Model and a Hyper-Network [9] to predict latent probabilities using two condition hyper-latents and previously decoded symbols. An autoregressive component was used for local dependencies and a hyperprior for global dependencies. The model achieved highly adaptive and efficient image compression. However, autoregressive decoding was sequential, which increased the complexity, and training and testing time. A discredited Gaussian mixture likelihood was used by [10] to model the probability distribution of the data, and introduce the anchor and attention models to focus on the complex and critical regions, which have more information and are hard to compress. These architectures enhance performance and allow the network to model complex dependencies and achieve superior reconstruction quality at low bitrates. However, this increases the model size and requires higher computational requirements for training. To reduce the complexity, [25] used a finite set of states to represent the possible outcomes of the autoregressive process, while [26] enhanced the hyperprior architecture using an adaptive skipping algorithm based on a threshold to skip the latent representations with a low impact on reconstruction quality and reduce the size of convolution kernels.

Image Transmission Over Wireless Networks As wireless communication technologies are rapidly changing, this affects how images are transmitted across networks and produces different challenges due to channel conditions interference, as well as increases the demand for efficient data compression, efficient channel coding, and error correction techniques. To address these challenges and optimise the transmission process, different communication systems have been developed. These systems can be divided into two categories: traditional communication systems, which include data encoding, channel encoding, channel and data decoding; and semantic communication systems, which focus on transmitting the essential meaning of images instead of sending the actual ground truth of the image. These communication systems are discussed below:

Traditional Communication Systems Several studies on traditional communication systems have investigated the effect of wireless channel errors on image transmission. The authors in [11] investigated and tested different compression techniques, such as run-length encoding, precision run-length encoding, DCT and DWT. Their study focused on addressing how errors affect the amount of data transmitted, without considering the effect on image quality. In a related study, [12] examined different types of data (text, audio and image) transmitted via an orthogonal frequency division multiplexing (OFDM) system along with the maximum likelihood detection technique and analysed different mapping techniques, such as binary phase-shift keying (BPSK), 4-level quadrature amplitude modulation, 16-level quadrature amplitude modulation (16-QAM), 32-level quadrature amplitude modulation (32-QAM) and 64-level quadrature amplitude modulation (64-QAM). This study addressed how errors affect the channel in two terms: bit error rate (BER) and mean squared error (MSE). Using machine learning with OFDM showed better performance in text transmission. Moreover, the schemes with a lower modulation order performed better than the others. Another study [13] investigated the behaviour of image transmission over Rayleigh fading channels using the OFDM technique. It tested the per-

formance using different modulation techniques, such as BPSK, 16-QAM, 32-QAM and 64-QAM. Similarly, [14] investigated the effect of wireless channel errors on steganographic images and proposed a system for steganographic images, which are images with securely hidden information. This system is based on DCT and DWT and encrypts the data using a chaotic Baker map. In addition, the authors in [15] proposed an OFDM multi-user massive multiple-input multiple-output system that uses fast Fourier transform, fractional Fourier transform and DWT transforms. They tested the system with different modulations, such as BPSK, quadrature phase-shift keying (QPSK), 8-PSK, 16-PSK, 32-PSK and 64-PSK, over a Rayleigh channel and found that the image quality decreases when the number of users increases.

Semantic Communication Systems Recently, semantic communication systems have gained the attention of many researchers. The authors in [16] developed a semantic communication system for image transmission in which the sender extracts the semantic map and then sends it instead of the actual ground truth of the image. On the receiver side, it uses a pre-trained deep learning GAN to produce the output image using the existing knowledge base. This knowledge is shared between the encoder and the decoder. This approach aims to save bandwidth by transmitting the compressed semantic map instead of the full image. In addition, a semantic communication method based on compression ratio optimisation was proposed to effectively deal with vision tasks [17]. This method conducts a semantic conversion of image data on the sender's internet of things (IoT) device through three stages, namely feature extraction, semantic relationship extraction and semantic compression, thus effectively reducing the total amount of data and transmission delay during communication. This method optimises the compression ratio to meet the requirements of computer vision tasks, successfully decreasing the data while maintaining key features. A flexible simulation software proposed by [18] transmits semantic segmentation map images using polar codes. The proposed software allows users to select the coding parameters and the SNR. Moreover, [19] produced an E2E adaptive semantic communication coding scheme that uses probabilistic graphical methods to extract and interpret the hierarchical semantic structure within images. The authors utilised a hierarchical parsing tree for unsupervised semantic representation, an enhanced encoder with cross-attention for faster and more accurate image characterisation and an adaptive encoding strategy that dynamically adjusts the bitrate based on semantic complexity. The adaptive coding strategy aims to optimise performance to achieve superior results, even at reduced bitrates.

3 Methodology

Wireless channel errors can occur at any stage during the transmission of data. The evolution of wireless technologies affects how images are transmitted and increases the need for efficient compression and error correction strategies to maintain visual quality. Some studies have used traditional communication systems to evaluate channel errors that transmit raw or compressed pixel data, disregarding the semantics of the image content, as in [14]. Conversely, studies that have developed semantic communication systems have used NNs to extract features and send these image features. For example, in [17], only semantic maps reconstructed by GANs are sent to the receiver. This reduces the bandwidth, as only compact semantic representations are transmitted, which is useful in low-capacity networks. Each component of the traditional and semantic communication systems is separated. A new E2E system was introduced using joint source-channel coding (JSCC) by applying

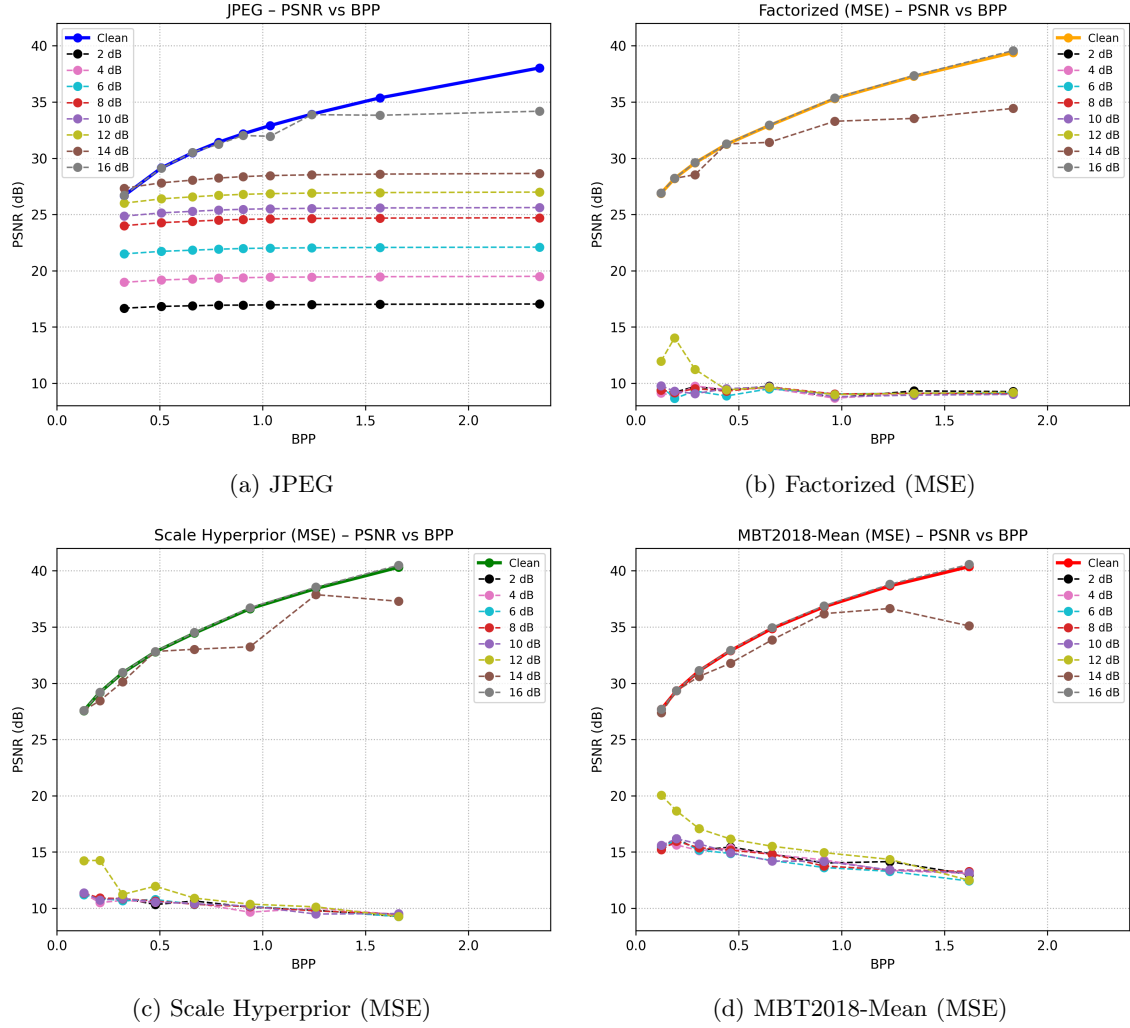


Fig. 2: PSNR vs BPP curves for different codecs under a clean channel and various noise levels (SNR 2–16 dB) in the Kodak dataset. JPEG gradually increases in PSNR when the channel noise decreases. However, all LBCs decrease in PSNR when the channel noise increases.

two CNNs for encoding and decoding and directly sending the output from the encoder through the channel [27].

In this study, we examine the robustness of learning-based image compression models. These models are similar to the framework used in JPEG AI, which compresses images into bitstreams and transmits them directly over the channel. In this case, using a traditional communication system is more appropriate and allows us to analyse how channel noise corrupts the transmitted bitstream and the effect on the visual quality of reconstructed images across different SNR levels. Semantic communication systems and JSCC are not suitable because they transmit only high-level features and do not convert them into actual bitstreams. The analysis focuses on the effect of channel noise on the received bitstream and the visual quality of the reconstructed images under varying SNR levels.

3.1 Simulation Setup

The communication system consisted of several components: a transmitter, a modulator, an additive white Gaussian noise (AWGN) channel, a demodulator and a receiver. The transmitter used source coding to encode the input image into bitstreams, which were divided into packets of size 1024 bytes, and modulate it using QPSK. The modulated packets were transmitted through the AWGN channel. On the receiver side, the packets were demodulated using a QPSK demodulator. The bitstream was passed to the decoder to recover the final image. The simulation was run for SNR values of 2–16 dB in steps of 2 dB. For each SNR level, the quality of the reconstructed image was evaluated using the peak signal-to-noise ratio (PSNR) and multi-scale structural similarity (MS-SSIM). To ensure accurate performance evaluation, the scenario was run multiple times across the entire Kodak dataset, and the results were averaged over multiple transmissions.

3.2 Tested Learning-Based Image Coding Models

This study assessed three different E2E learning-based image compression models and included standard JPEG as a reference codec for baseline comparison. The LBC models evaluated in this work were chosen from available CompressAI models because they represent different levels of side information, such as no side information (Factorised entropy model (bmshj2018-factorised) [7]), a hyperprior as side information (Scale Hyperprior with zero-mean Gaussian conditionals [7]) and side information with mean and variance Gaussian conditionals (Scale Hyperprior with non-zero-mean Gaussian conditionals (MBT2018-Mean) [9]). All were optimised for MSE.

4 Results and Analysis

In our experiments, we consider two channel conditions: clean (error-free channel) and noisy transmission conditions.

4.1 Rate-Distortion Performance

To analyse the R-D, the images are compressed using the selected image codec in various quality factors, and the coded images are decompressed. The quality of the reconstructed images is assessed using objective performance metrics (e.g., PSNR and MS-SSIM). In this study, all 24 high-quality uncompressed images from the Kodak dataset [21] are used in the experiments, and the average values of the performance metrics are calculated. Fig. 2 shows the R-D curve (labelled "Clean" in the legends) for different image compression techniques. The standard JPEG shows the lowest PSNR values compared with the LBC models. The LBC outperformed the traditional JPEG codec across the entire range of bitrates. Among the AI models, MBT2018-Mean (MSE) achieves the highest PSNR values as it uses the mean and variance to predict the distribution. Fig. 3 illustrates the MS-SSIM (dB) (labelled "Clean" in the legends) against BPP. All learned methods achieve excellent perceptual quality. MBT2018-Mean and Scale Hyperprior show similar results, delivering the highest MS-SSIM values and enhancing compression while preserving image quality.

4.2 Effect of Wireless Channel Errors on Visual Quality

To investigate AI image coding error behaviour, the AI models are tested under varying noise conditions to analyse error patterns and assess their robustness. The SNR is varied

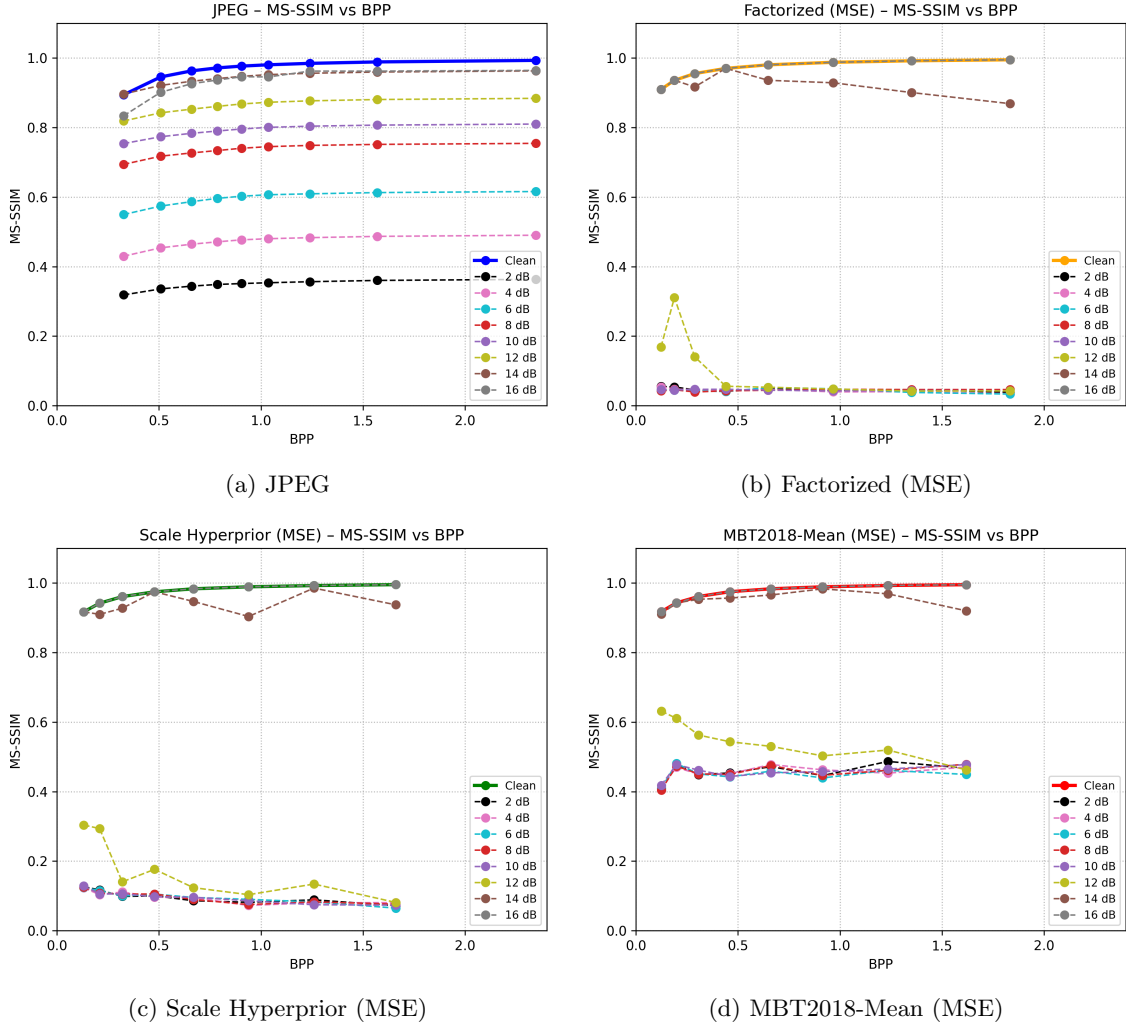


Fig. 3: MS-SSIM vs. BPP curves for different codecs under a clean channel and various noise levels (SNR 2–16 dB) in the Kodak dataset. JPEG gradually increases in MS-SSIM when the channel noise decreases. However, all LBCs decrease in MS-SSIM when the channel noise increases.

across a range of values to achieve different levels of BER. First, the images are compressed with different quality factors using the selected image codecs. Second, compressed images are transmitted through the wireless channel with varying levels of SNR. Third, after receiving the data, the compressed images are decoded, and the average performance metrics (PSNR and MS-SSIM) are computed across the entire testing dataset. The results in Figs. 2 and 3 show that increasing the SNR gradually increases the performance in terms of PSNR and MS-SSIM for the traditional method.

The AI models have four cases:

1. (SNR < 12) All methods show significant degradation in PSNR (less than 16 dB) and MS-SSIM (less than 0.5 dB), regardless of bitrate. This means that these methods are sensitive to channel errors.
2. (SNR = 12) At a low bitrate, the MBT2018-Mean shows a small improvement in the results in PSNR and MS-SSIM compared with the two other models.

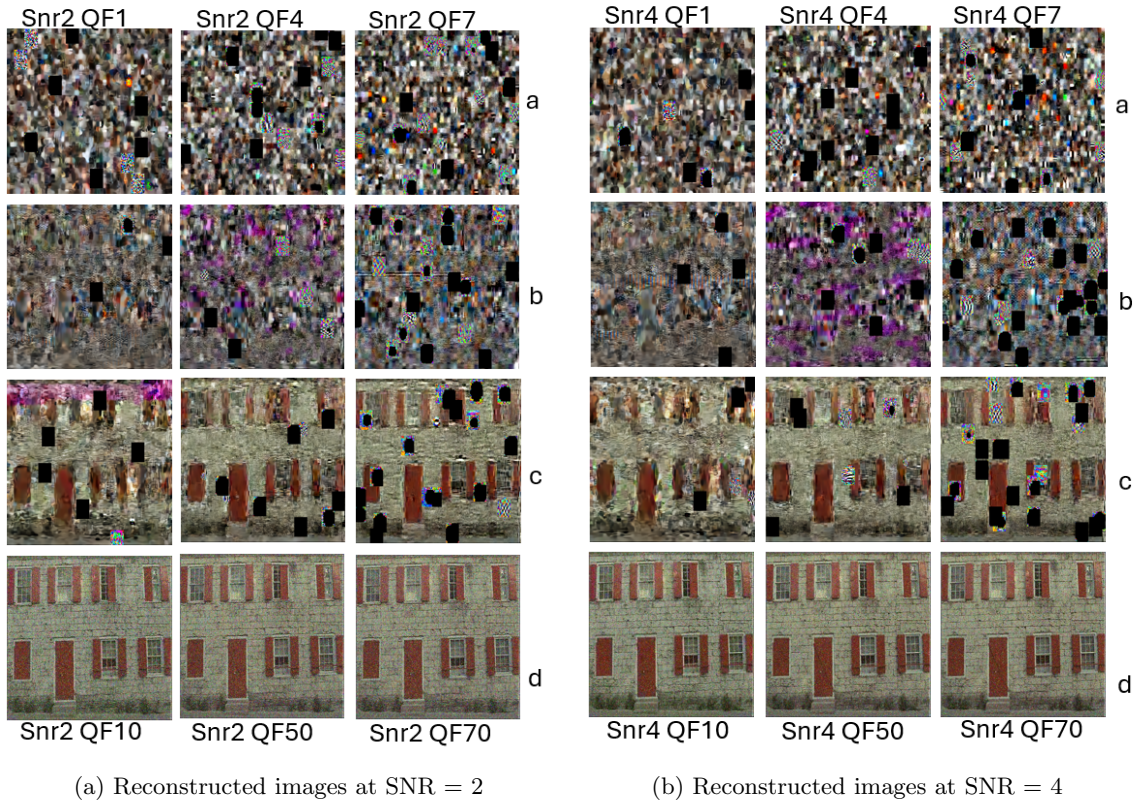


Fig. 4: Examples of reconstructed images using different methods. (a) Factorised, (b) Scale Hyperprior, (c) MBT2018-Mean and (d) JPEG, with different quality factors (QFs) and low SNR values.

3. (SNR = 14,) All methods show a notable improvement in PSNR (higher than 26 dB) and MS-SSIM (higher than 0.8 dB), especially at low bitrates.
4. (SNR > 14) The AI models have similar performance in clean and noisy channels.

In the noise channel, JPEG successfully reconstructs images with some missing pixels and a blurry artefact, as shown in Figs. 4 - 6. The reconstructed image quality improves as the SNR increases.

For the AI models, the Factorised model fails to reconstruct the images and shows corrupted images, as shown in Figs.4 and 5, because any errors in the bitstreams may propagate and the entropy coder is unable to decode the bitstreams. In Fig.6, the model reconstructs part of the image when the SNR is 12 at low quality, which leads to corrupted regions. However, the model fails to reconstruct the entire image when the QF is high. At above SNR 14, the model shows results similar to the no-noise case; this is because the model can remove the compressed artefacts and conceal the errors when the signal quality is high enough.

The Scale Hyperprior shows the worst results and fails to reconstruct the image when the SNR is below 14, as shown in Fig.4, 5 and 6. Even when the signal quality is enhanced (SNR=14), it fails when the QF is 4 (Fig.6) because of corruption and error propagation in the side information.

The mean scale hyperprior used in MBT2018-Mean reduces the error effect because it reduces the model-distribution mismatch. However, this may cause colour artefacts or missing parts, as shown in Figs.4 and 5. When the signal quality improves (above SNR

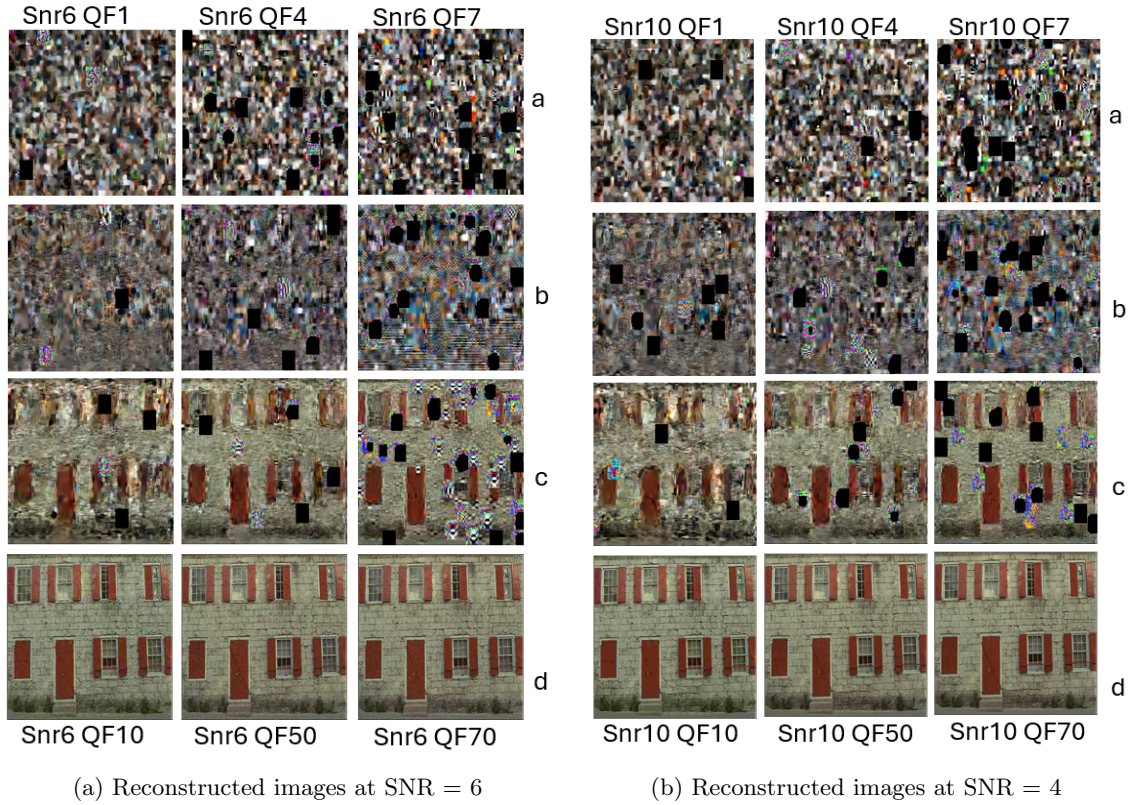


Fig. 5: Examples of reconstructed images using different methods. (a) Factorised, (b) Scale Hyperprior, (c) MBT2018-Mean and (d) JPEG, with different quality factors (QFs) and moderate SNR values.

14), the model removes compressed artefacts and conceals errors.

To understand the robustness of each method to the channel noise, for each SNR level, the PSNR and MS-SSIM performance are calculated in all dataset images across all compression levels (i.e.QFs). This shows the expected performance across all compression levels. The averaged PSNR (Fig. 7) and the averaged MS-SSIM (Fig. 8) illustrate the overall noise robustness of each method. JPEG shows a gradual and steady increase in performance as SNR increases. The AI methods show better quality only in high SNR values (above 12 dB). However, these models are less resilient to noise, resulting in low PSNR and MS-SSIM in degraded channels. The error bars represent the standard deviation of PSNR and MS-SSIM in all Kodak images and all tested compression rates (i.e.QFs) for each SNR level. At low SNRs, the error bars are relatively small, which means consistent degradation across all compression levels. Regardless of the QFs, if any error occurs, the error is propagated, and the range Asymmetric Numeral Systems (rANS) fails to recover the correct compact presentation. At high SNRs, the error bars increase, which reflects greater variability, depending on the bitrate and image content.

5 Discussion

The main difference between the AI models used in the experiments is how they model the probability distributions of the latent variables. The Factorized model uses a simple

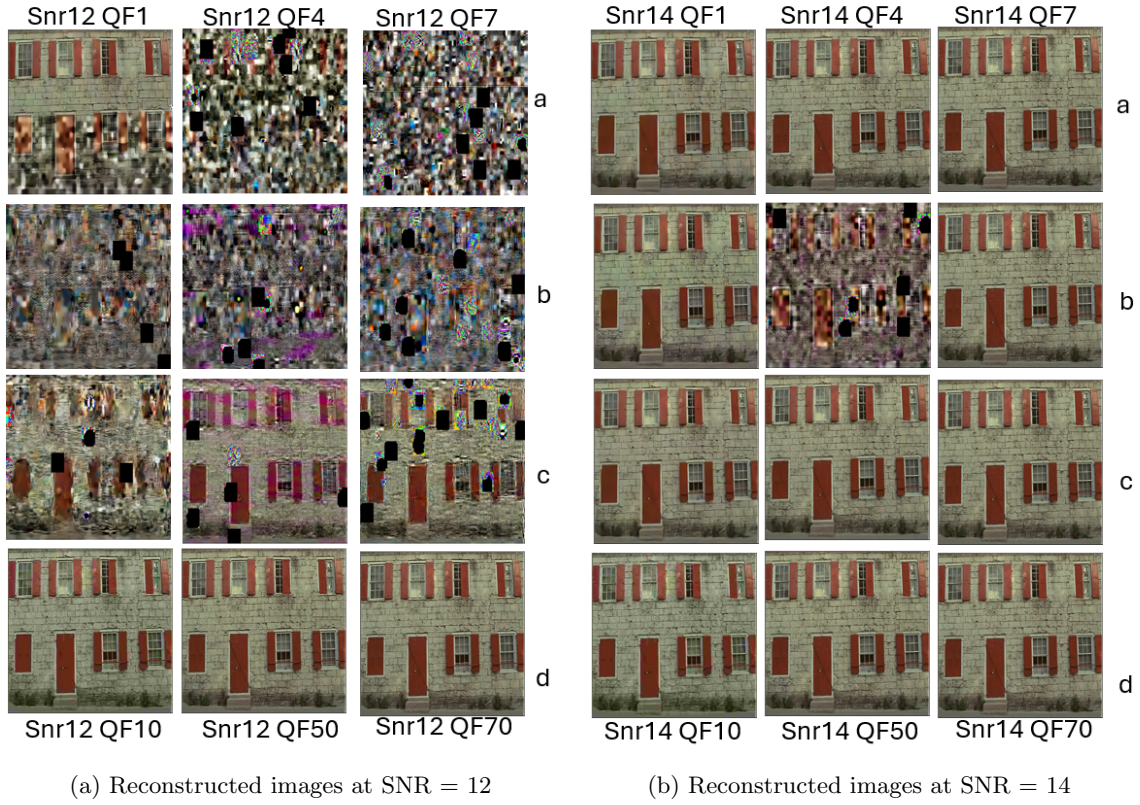


Fig. 6: Examples of reconstructed images using different methods. (a) Factorised, (b) Scale Hyperprior, (c) MBT2018-Mean and (d) JPEG, with different quality factors (QFs) and high SNR values.

independent entropy model that assumes that all latents are identical as well as a Gaussian distribution without any side information. The Scale Hyperprior model improves compression by introducing a separate hyperprior network that assumes zero means and predicts scale parameters (standard deviations) for each latent. This helps the model predict more accurate probabilities and enhances its performance. The MBT2018-Mean further enhances it by predicting the mean and the scale of each latent variable conditioned on the hyperprior. All three methods use rANS [20] for entropy coding to achieve high compression efficiency and reduce the number of bits needed to represent the data. These models outperform the traditional method in terms of PSNR and MS-SSIM performance, while preserving image quality at low bitrates. However, these models fail to reconstruct the image as shown in Figs.4, Fig.5, and Fig.6, and often cause severe artifacts or the whole decoding fails if any error occurs. Corruption and error propagation in the side information increase the artefacts. This error propagation is a major challenge when using rANS-coded latent representations over unreliable wireless channels.

The MBT2018-Mean model performs better; it only shows some missing regions and blurry artefacts, as the use of learned mean and variance reduces the mismatch in the model distribution. There is no block-based processing in these models; instead, they treat the whole image as a single block and convert it into bitstreams using rANS. The rANS compresses the data depending on the previous symbols or bits in the stream. This explains why the AI models suddenly improve their performance when the channel conditions improve (after SNR 12 dB). By contrast, JPEG shows a smoother and gradual

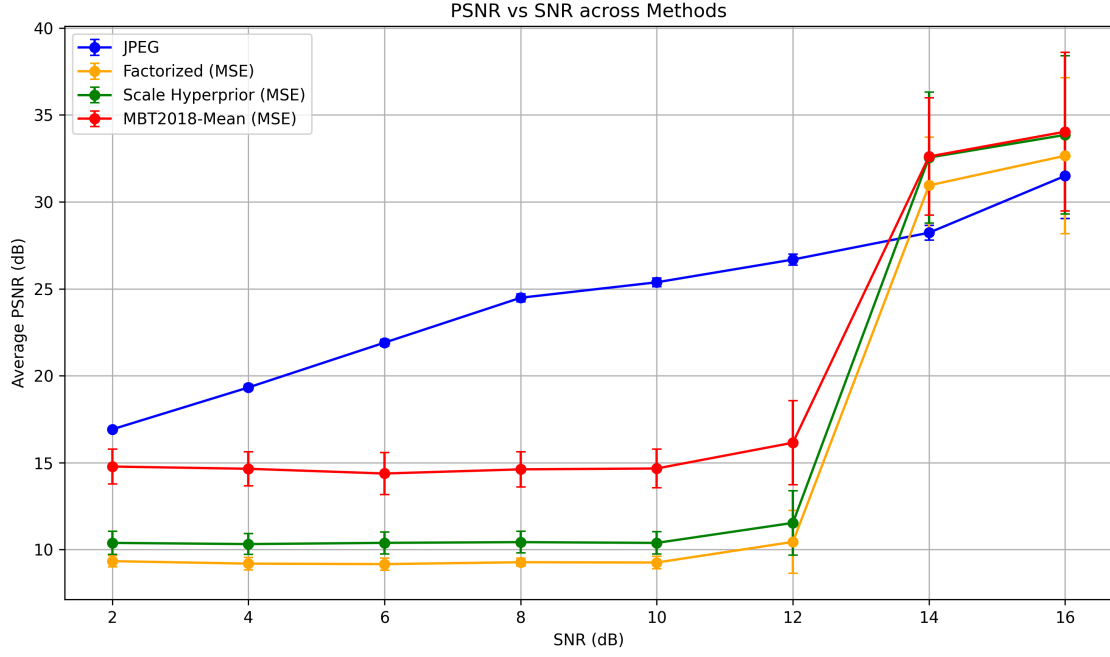


Fig. 7: Average PSNR vs. SNR for the codec methods. Each curve shows the mean PSNR in all Kodak dataset images and compression rates. The error bars represent the standard deviation of the PSNR across images and quality values at each SNR level.

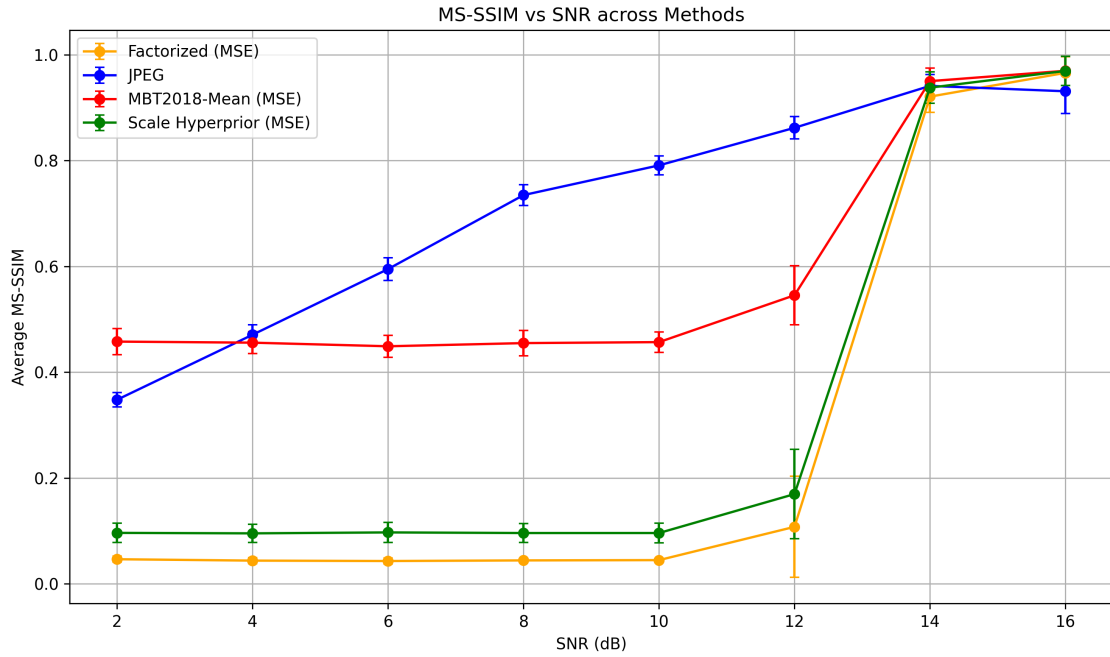


Fig. 8: Average MS-SSIM vs. SNR for the codec methods. Each curve shows the mean MS-SSIM in all Kodak dataset images and compression rates. The error bars represent the standard deviation of the MS-SSIM across images and quality values at each SNR level.

enhancement in reconstruction quality as the channel conditions improve. JPEG is more

resilient to errors and can reconstruct images even when the SNR is low. Its robustness comes from encoding the image in independent blocks. This block-based design localises the errors in specific regions and prevents error propagation. Another reason is that JPEG uses fixed and simple statistical assumptions and does not adapt a model during encoding. This makes its bitstreams less sensitive to small errors, as decoding does not depend on learning statistical dependencies. Another important point is that these AI models typically do not consider the channel part during training. Some models use noise to train the models to be robust to quantisation errors, but not to real channel noise. New E2E communication systems, such as [27], have opened a new era to explore the use of real E2E training frameworks.

6 Conclusions

This study evaluated the rate-distortion performance and robustness of several AI-based image compression methods compared with the traditional JPEG codec under different wireless channel conditions. The results demonstrated that LBCs, such as Factorized, Scale Hyperprior, and MBT2018-Mean, significantly outperformed JPEG in terms of compression efficiency and reconstructed image quality at high SNR levels. However, they showed high sensitivity to bit errors introduced by channel noise. By contrast, JPEG, despite offering lower compression efficiency, demonstrated superior resilience to errors, with a gradual degradation in image quality as the SNR decreased. The LBCs showed a sharp degradation in quality when the SNR dropped below 14 dB. This is mainly because they use rANS entropy coding, which causes even small errors to propagate through the entire bitstream, making it impossible to recover the image partially. Future work should focus on improving the error resilience for LBC models, especially for applications in a noisy channel. One critical direction is to develop advanced error concealment mechanisms that can reduce the visual effects of bitstream corruption during decoding. Error propagation must be restricted and the effect must be minimised by using a block-based architecture or a chunk mechanism for the image or the bitstreams. More robust entropy coding methods that use local and global context modelling must be developed to control error propagation and limit its effect. New and fast methods of context modelling must also be developed, as sequential context modelling could be slow and unsuitable for real-time applications. Furthermore, most LBC models are typically trained with simple noise during quantisation or under the assumption of clear channel conditions, which does not reflect the real transmission scenarios. Researchers should consider training their models with realistic channel components embedded within the learning process. Recent progress in learning-based communication systems offers a promising path forward. Through integration, improved entropy coding, error propagation control, and channel-aware training, future learning-based image codecs could significantly reduce the gap between high compression performance and reliable transmission, enabling practical deployment in limited bandwidth and error-prone environments.

References

1. E. Khorov, A. Kiryanov, A. Lyakhov, and G. Bianchi, "A tutorial on IEEE 802.11 ax high efficiency WLANs," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 197–216, 2018.
2. G. Tiwari, D. Nandi, and M. Mishra, "A comparative study on image and video compression techniques," *IOSR Journal of VLSI Signal Processing*, vol. 3, no. 3, pp. 69–73, 2013.
3. Joint Photographic Experts Group, "JPEG Standard." Available online: <https://www.jpeg.org/jpeg/> Accessed: April 14, 2025.

4. T. Acharya and P.-S. Tsai, *JPEG2000 Standard for Image Compression: Concepts, Algorithms and VLSI Architectures*. John Wiley & Sons, 2005.
5. P. Eusebio, J. Ascenso, and F. Pereira, "Optimizing an Image Coding Framework with Deep Learning-based Pre- and Post-Processing," in *Proc. 2020 EUSIPCO*, pp. 506–510, 2021.
6. I. Schiopu, Y. Liu, and A. Munteanu, "CNN-based prediction for lossless coding of photographic images," in *Proc. 2018 Picture Coding Symposium (PCS)*, pp. 16–20, 2018.
7. J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, "Variational image compression with a scale hyperprior," *arXiv preprint arXiv:1802.01436*, 2018.
8. ISO/IEC JTC 1/SC29/WG1, "White Paper on JPEG AI Scope and Framework v1.0," *ISO/IEC JTC 1/SC29/WG1 N90049*, 2021.
9. D. Minnen, J. Ballé, and G. D. Toderici, "Joint autoregressive and hierarchical priors for learned image compression," in *Advances in Neural Information Processing Systems*, vol. 31, 2018.
10. Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Learned image compression with discretized Gaussian mixture likelihoods and attention modules," in *Proc. IEEE/CVF CVPR*, pp. 7939–7948, 2020.
11. M. Chandra, D. Agarwal, and A. Bansal, "Image transmission through wireless channel: A review," in **Proc. IEEE 1st Int. Conf. Power Electron., Intell. Control and Energy Syst. (ICPEICES)**, 2016, pp. 1–4.
12. A. Y. Reddy, B. L. Reddy, A. S. N. V. Sai, K. Anuraj, *et al.*^{*}, "MSE and BER analysis of text, audio and image transmission using ML based OFDM," in **Proc. IEEE Int. Conf. Innovation in Technol. (INOCON)**, 2020, pp. 1–3.
13. M. Chandra, D. Agarwal, and A. Bansal, "Performance analysis of image transmission through Rayleigh channel," in **Proc. 8th Int. Conf. Comput., Commun. and Netw. Technol. (ICCCNT)**, 2017, pp. 1–5.
14. A. A. Eyssa, F. E. Abdelsamie, and A. E. Abdelnaiem, "An efficient image steganography approach over wireless communication system," **Wireless Pers. Commun.**, vol. 110, no. 1, pp. 321–337, 2020.
15. L. Kansal, S. Berra, M. Mounir, R. Miglani, R. Dinis, and K. Rabie, "Performance analysis of massive MIMO-OFDM system incorporated with various transforms for image communication in 5G systems," **Electronics**, vol. 11, no. 4, p. 621, 2022.
16. M. U. Lokumarambage, V. S. S. Gowrisetty, H. Rezaei, T. Sivalingam, N. Rajatheva, and A. Fernando, "Wireless end-to-end image transmission system using semantic communications," **IEEE Access**, 2023.
17. Z. Wan, S. Liu, Z. Xu, W. Ni, Z. Chen, and F. Wang, "Semantic communication method based on compression ratio optimization for vision tasks in the artificial intelligence of things," **IEEE Trans. Consum. Electron.**, 2023.
18. H. Rezaei, T. Sivalingam, and N. Rajatheva, "Automatic and flexible transmission of semantic map images using polar codes for end-to-end semantic-based communication systems," in **Proc. IEEE 34th Annu. Int. Symp. Pers., Indoor and Mobile Radio Commun. (PIMRC)**, 2023, pp. 1–6.
19. G. Shi, H. Li, D. Gao, M. Yang, and Y. Dong, "An image adaptive rate mechanism in semantic communication for image endogenous semantics," **IEEE Trans. Veh. Technol.**, 2024.
20. InterDigital, "CompressAI: A PyTorch library and evaluation platform for end-to-end compression research." [Online]. Available: <https://interdigitalinc.github.io/CompressAI/>. Accessed: July. 2, 2025.
21. D. Knuth, "Kodak Lossless True Color Image Suite (PhotoCD PCD0992)." [Online]. Available: <http://r0k.us/graphics/kodak/>. Accessed: May. 20, 2025.
22. Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Deep convolutional autoencoder-based lossy image compression," in *Proc. 2018 PCS*, pp. 253–257, 2018.
23. A. Prakash, N. Moran, S. Garber, A. DiLillo, and J. Storer, "Semantic perceptual image compression using deep convolution networks," in *Proc. 2017 DCC*, pp. 250–259, 2017.
24. E. Agustsson, M. Tschannen, F. Mentzer, R. Timofte, and L. Van Gool, "Generative adversarial networks for extreme learned image compression," in *Proc. IEEE/CVF ICCV*, pp. 221–231, 2019.
25. Y. Zhang, H. Yu, J. Li, and W. Lin, "Finite-State Autoregressive Entropy Coding for Efficient Learned Lossless Compression," in *Proc. Twelfth International Conference on Learning Representations*, 2024.
26. H. Zhang, F. Mei, J. Liao, L. Li, H. Li, and D. Liu, "Practical learned image compression with online encoder optimization," in *Proceedings of the 2024 Picture Coding Symposium (PCS)*, 2024, pp. 1–5.
27. E. Bourtsoulatzé, D. B. Kurka, and D. Gündüz, "Deep joint source-channel coding for wireless image transmission," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 3, pp. 567–579, 2019.